EP33952 (1)

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
28 March 2002 (28.03.2002)

PCT

(10) International Publication Number
WO 02/25575 A2

(51) International Patent Classification⁷: G06K 9/00

(21) International Application Number: PCT/US01/29768

(22) International Filing Date: 24 September 2001 (24.09.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/234,813	22 September 2000 (22.09.2000)	US
09/895,868	29 June 2001 (29.06.2001)	US
09/897,786	29 June 2001 (29.06.2001)	US

(71) Applicant: SRI INTERNATIONAL [US/US]; 333 Ravenswood Avenue, Menlo Park, CA 94024 (US).

(72) Inventors: MYERS, Gregory, K.; 473 Melrose Avenue, San Francisco, CA 94127 (US). GALLAGHER, Paul, K.; 15110 Herring Avenue, San Jose, CA 95124 (US). BOLLES, Robert, C.; 1465 Lloyd Way, Mountain View, CA 94040 (US). LUONG, Quang-Tuan; 786 Coleman Avenue #L, Menlo Park, CA 94025 (US). HERSON, James, A.; 735 Vernal Way, Emerald Hills, CA 94062 (US).

(74) Agent: KIN-WAH, Tong; Moser, Patterson & Sheridan, LLP, Suite 100, 595 Shrewsbury Avenue, Shrewsbury, NJ 07702 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

WO 02/25575 A2

(54) Title: METHOD AND APPARATUS FOR PORTABLY RECOGNIZING TEXT IN AN IMAGE SEQUENCE OF SCENE IMAGERY

(57) Abstract: An apparatus (100, 1200) and a concomitant method for detecting and recognizing text information in a captured imagery. The present method transforms the image of the text to a normalized coordinate system before performing OCR, thereby yielding more robust recognition performance. The present invention also combines OCR results from multiple frames, in a manner that takes the best recognition results from each frame and forms a single result that can be more accurate than the results from any of the individual frames. In one embodiment, the present invention is a portable device that is capable of capturing imagery and is also capable of detecting and extracting text information from the captured imagery. The portable device contains an image capturing sensor, a text detection module, an OCR module, a storage device and means for presenting the output to the user or other devices.

METHOD AND APPARATUS FOR PORTABLY RECOGNIZING TEXT IN AN IMAGE SEQUENCE OF SCENE IMAGERY

The present invention relates to an apparatus and concomitant method for digital image processing. More specifically, the present invention provides text recognition in an image sequence of scene imagery, e.g., three-dimensional (3D) scenes of the real world.

BACKGROUND OF THE DISCLOSURE

Video and scene imagery are increasingly important sources of information. The proliferation and availability of devices such as digital still cameras and digital video cameras are clear evidence of this trend.

Aside from the general scenery, e.g., people, and the surrounding landscape, many captured imagery often contain text information (e.g., broadly including letters, numbers, punctuation and symbols). Although the captured text information is easily recognizable by a human viewer, this important text information is often not detected and deciphered by the portable image capturing device and therefore is not immediately utilized by the operator of the portable image capturing device.

However, it has been noted that recognizing text that appears in real-world scenery is potentially useful for characterizing the contents of video imagery, i.e., gaining insights about the imagery. In fact, the ability to accurately deduce text information within real-world scenery will enable the creation of new applications that gather, process, and disseminate information about the contents of captured imagery.

Additionally, the volume of collected multimedia data is expanding at a tremendous rate. Data collection is often performed without real time processing to deduce the text information within the captured data. For example, captured imagery can be stored in a portable device, but no processing is performed to detect and extract text information within the captured imagery. Thus, benefits associated with real time text detection and extraction are not realized in portable imagery capturing devices.

Therefore, a need exists in the art for an apparatus and method to portably detect and extract text information from captured imagery, thereby allowing new implementations for the gathering, processing, and dissemination of information relating to the contents of captured imagery.

SUMMARY OF THE INVENTION

One embodiment of the present invention is an apparatus and a concomitant method for detecting and recognizing text information in video imagery or in a sequence of images. In one embodiment, the present invention transforms the image of the text to a normalized coordinate system before performing OCR, thereby yielding more robust recognition performance.

In a second embodiment, the present invention also combines OCR results from multiple frames, in a manner that takes the best recognition results from each frame and forms a single result that can be more accurate than the results from any of the individual frames.

In a third embodiment, the present invention is an apparatus and a concomitant method for portably detecting and recognizing text information in captured imagery. In the embodiment, the present invention is a portable device that is capable of capturing imagery and is also capable of detecting and extracting text information from the captured imagery. The portable device contains an image capturing sensor, a text detection module, an OCR module, and means for presenting the output to the user or other devices. Additional modules may be necessary for different embodiments as described below.

In a fourth embodiment, the present device is deployed as a portable language translator. For example, a user travelling in a foreign country can capture an imagery having text (e.g., taking a picture of a restaurant menu). The text within the captured imagery is detected and translated to a native language of the user. A pertinent language translator can be loaded into the portable device.

In a fifth embodiment, the present device is deployed as a portable assistant to an individual who is visually impaired or who needs reading assistance. For example, a user shopping in a store can capture an imagery having text (e.g., taking a picture of the label of a product). Another example is a child taking a picture of a page in a book. The text within the captured imagery is detected and audibly broadcasted to the user via a speaker.

In a sixth embodiment, the present device is deployed as a portable notebook. For example, a user in an educational environment can capture an imagery having text (e.g., taking a picture of a white board, view graph or a screen). The text within the captured imagery is detected and stored in a format that can be retrieved later for text processing, e.g., in a word processor format.

In a seventh embodiment, the present device is deployed as a portable auxiliary information accessor. For example, a user in a business environment can capture an imagery having text (e.g., taking a picture of a billboard or a business card having an Internet or web address). The text within the captured imagery is detected and the Internet address is accessed to acquire additional information.

In an eighth embodiment, the present device is deployed as a portable navigation assistant. For example, the portable unit is deployed in a vehicle for automatic reading of road signs and speed limit signs. The text within the captured imagery is detected and is provided to the computer in the vehicle for assisting the vehicle's navigation system or as a warning indicator to the driver on an instrument panel.

In a ninth embodiment, the present device is deployed as a portable law enforcement assistant. For example, the portable unit is deployed in a police vehicle or in a hand-held device for reading license plates, vehicle identification numbers (VINs) or driver licenses and registrations. The text within the captured imagery is detected and is used to provide information to a law enforcement officer as to the status of a vehicle or a driver.

In a tenth embodiment, the present device is deployed as a portable inventory assistant. For example, a user in a store or a warehouse can capture an imagery having text (e.g., taking a picture of a product on a shelf or high up on a scaffold). In another example, the odometer reading for a returned rental car could be automatically captured. The text within the captured imagery is detected and is used for inventory control.

BRIEF DESCRIPTION OF THE DRAWINGS

The teachings of the present invention can be readily understood by considering the following detailed description in conjunction with the accompanying drawings, in which:

FIG. 1 illustrates a block diagram of a text recognition system of the present invention;

FIG. 2 illustrates a method of text recognition and extraction in accordance with the text recognition system of the present invention;

FIG. 3 illustrates the orientation angle of text relative to a camera can be modeled in terms of three angles;

FIG. 4 illustrates text that appears to be rotated in the image plane;

FIG. 5 illustrates characters becoming sheared after in-plane rotation;

FIG. 6 illustrates a test image of a poster containing text that was captured at an azimuth angle of 70 degrees;

FIG. 7 illustrates a method for estimating top line and bottom line;

FIG. 8 illustrates an example of shear computation;

FIG. 9 illustrates refined bounding boxes based on the top and base lines and on the dominant vertical direction;

FIG. 10 illustrates the warped text lines after the baseline refinement and deshearing;

FIG. 11 illustrates a 70 degree azimuth test image, with the recognition results overlaid on the normalized image;

FIG. 12 illustrates a block diagram of a portable text recognition system of the present invention;

FIG. 13 illustrates a method of utilizing the portable text recognition system of the present invention in a first embodiment;

FIG. 14 illustrates a method of utilizing the portable text recognition system of the present invention in a second embodiment;

FIG. 15 illustrates a method of utilizing the portable text recognition system of the present invention in a third embodiment;

FIG. 16 illustrates a method of utilizing the portable text recognition system of the present invention in a fourth embodiment;

FIG. 17 illustrates a method of utilizing the portable text recognition system of the present invention in a fifth embodiment;

FIG. 18 illustrates a method of utilizing the portable text recognition system of the present invention in a sixth embodiment; and

FIG. 19 illustrates a method of utilizing the portable text recognition system of the present invention in a seventh embodiment.

To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common to the figures.

DETAILED DESCRIPTION

FIG. 1 illustrates a block diagram of a text recognition device or system 100 of the present invention. In one embodiment, the text recognition device or system 100 is implemented using a general purpose computer or any other hardware equivalents. Although the recognition device or system 100 is preferably implemented as a portable device, it should be noted that the present invention can also be implemented using a larger computer system, e.g., a desktop computer or server.

Thus, text recognition device or system 100 comprises a processor (CPU) 130, a memory 140, e.g., random access memory (RAM) and/or read only memory (ROM), a text recognition and extraction engine 120, and various input/output devices 110, (e.g., storage devices 111, including but not limited to, a tape drive, a floppy drive, a hard disk drive or a compact disk drive, a receiver 112, a transmitter 113, a speaker 114, a display 115, an image capturing sensor 116, e.g., digital still camera or video camera, a keyboard, a keypad, a mouse, and the like).

It should be understood that the text recognition and extraction engine 120 can be implemented as physical devices that are coupled to the CPU 130 through a communication channel. Alternatively, the text recognition and extraction engine 120 can be represented by one or more software applications (or even a combination of software and hardware, e.g., using application specific integrated circuits (ASIC)), where the software is loaded from a storage medium, (e.g., a magnetic or optical drive or diskette) and operated by the CPU in the memory 140 of the computer. As such, the text recognition and extraction engine 120 (including associated methods and data structures) of the present invention can be stored on a computer readable medium, e.g., RAM memory, magnetic or optical drive or diskette and the like.

The text recognition and extraction engine 120 comprises a text detection module 121, a text orientation module 122, a text binarization module 123, an optical character recognition (OCR) module 124, an agglomeration module 125, a lexicon processing module 126, and a false text detection module 127. In operation, the text recognition and extraction engine 120 is able to accurately detect and extract text information from an input image or video imagery. A detailed description of the functions of the text recognition and extraction engine 120 is disclosed below with reference to FIG. 2. The text results from the text recognition and extraction engine 120 is then provided to the processor 130 for further processing to provide various functionalities or services. These functionalities or services include, but are not limited to, automatic text recognition with audio playback or visual display (e.g., street signs, placards, restaurant menus, billboards, white boards, labels, or books), automatic text translation to a foreign language, automatic access to auxiliary information, automatic road sign reading for navigation, automatic license plate reading for law enforcement functions, image and video indexing and archiving and inventory and shelf restocking control. Examples of such portable text recognition devices are disclosed in US patent application entitled “Method And Apparatus For Portably Recognizing Text In An Image Sequence Of Scene Imagery” with attorney docket SRI/4314-2, which is herein incorporated by reference and is filed simultaneous herewith.

FIG. 2 illustrates a method 200 for text recognition and extraction in accordance with the present invention. Specifically, the method is designed for detecting and reading text appearing in video or still imagery. This invention is concerned with reading text that is in the scene itself (such as a sign on a building). However, the present techniques could also be applied to computer-generated text caption overlays (such as that which appears in broadcast news videos). The system 100 of FIG. 1 employing method 200 can accept a video or still image signal and recognize text in real time. It should be noted that the term “captured imagery” in the present application may encompass, in part or in whole, a still image and/or an image sequence.

Method 200 starts in step 205 and proceeds to step 210, where an image or an image sequence (e.g., video) is captured via conventional equipment, e.g., image capturing sensor 116 (e.g., digital still camera or digital video camera). Alternatively, step 210 can be omitted if the captured imagery was previously captured and is simply being retrieved from a storage device 111.

In step 220, method 200 detects, locates, and tracks text regions within the captured imagery. More specifically, method 200 approaches text detection and location with the assumption that the text is roughly horizontal, and that the characters have a minimum contrast level compared with the image background. The text may be of either polarity (light text on a dark background, or dark text on a light background). The method first detects vertically oriented edge transitions in the gray-scale image, using a local neighborhood edge operator. The output of the operator is thresholded to form two binary images, one for dark-to-light transitions (B1), and the other for light-to-dark transitions (B2). A connected components algorithm is applied on each binary image. The connected components that have been determined (by examining their height and area) not due to text are eliminated. The remaining connected components are linked to form lines of text by searching the areas to the left and right of each connected component for additional connected components that are compatible in size and relative position. Finally, a rectangle is fitted to each line of detected text or a group of lines of text using the moments of all connected components used to locate the text. Tracking text over multiple consecutive video frames is achieved by computing the frame-to-frame displacement of the pixels in a set of local neighborhoods, or finding and following distinctive shape features, such as the ends of character strokes, and then computing a geometric transformation that characterizes the frame-to-frame displacement of corresponding text regions.

Returning to FIG. 2, in step 230, method 200 adjusts detected text to account for orientation. Namely, text in a captured imagery is often viewed from an oblique angle as shown in FIG. 4 below. Such a configuration is quite common when the main subject of the scene is not the text itself, but such incidental text could be quiet important (for example, it may be the only clue of the location of the captured imagery). Thus, method 200 applies a processing step in step 230 to account for text orientation, thereby improving the OCR method that will be applied at a later processing stage. A detailed description of step 230 is provided below.

In step 240, method 200 binarizes the detected text regions. Binarization is performed on each text line independently. It is assumed that the text pixels are relatively homogeneous, and that the intensity of the background pixels may be highly variable. For each text line, the polarity of text is first determined, and then binarization of the text line is performed.

The polarity is determined by comparing grayscale pixel values above and below the baselines. This relies on the inside pixels (those below the top and above the bottom baselines) most likely being character pixels and the outside pixels (those above the top and below the bottom baseline) most likely being background pixels. The polarity calculation compares pairs of pixels along both baselines and sums the number of times the inside pixel is greater than the outside pixel. If this sum is greater than zero, the polarity is determined to be light text on a dark background; otherwise, the polarity is determined to be dark text on a light background.

In binarization, the gray-scale image is smoothed with a Gaussian kernel, and histograms H1 and H2 are computed. Histogram H1 is composed of gray-scale pixels in the smoothed image on the right side of the connected components in the dark-to-light edge transition image B1 and on the left side of the light-to-dark edge transition image B2. If light text is in this text region, these are the pixels most likely to belong to light text or near the edge of light text. Similarly, histogram H2 is composed of gray-scale pixels in the smoothed image on the right side of the connected components in image B2 and on the left side of the image B1. The threshold for the text line is then set to the gray value at the 60th percentile of histogram H1 or H2, depending on the polarity chosen. Alternatively, more than one binarizaton result for each text line is produced, each using a different threshold value (e.g., 45th percentile, 60th percentile, and 75th percentile). Producing more than one binarization result, and sending them through the OCR process (Step 250) can, after combining the OCR results with agglomeration (Step 260), sometimes yield more accurate results than processing a single binarization result.

Returning to FIG. 2, in step 250, method 200 applies OCR processing to the binarized text regions. In one embodiment, step 250 is achieved by using a commercially available OCR engine, e.g., OCR package from Scansoft, Inc. of Peabody, MA. However, it should be noted the present invention is not so limited and that other OCR packages may also be used. It should be noted that some OCR engines operate on a gray-scale imagery instead of binary images and therefore would not require the processing in step 240. The OCR engine produces one or more candidate identities for each recognized text character in the image, rank-ordered according to likelihood.

In step 260, method 200 agglomerates the OCR results. Specifically, a video text recognition process usually involves performing optical character recognition (OCR) on images derived from individual video frames. However, in many applications the same text persists in the scene for some length of time. Digitized video frames of the same scene may vary slightly, thereby causing an OCR process operating on individual frames to produce slightly different results. Therefore, method 200 combines ("agglomerates") OCR results from multiple frames, in a manner that takes the best recognition results from each frame and forms a single result. The use of agglomeration improves the recognition accuracy over that of the OCR results on individual images. It also enables the system to avoid outputting the same results repeatedly when the text is persistent in the video sequence for many frames, and reduces the generation of false characters from non-text image regions. In addition, because the agglomeration process works on OCR results (as opposed to image pixels) from multiple frames, it is computationally fast enough to implement in a real-time system (i.e., one that keeps up with the video display rate).

In general, method 200 uses the frame-to-frame displacement determined in Step 220 to find character-to-character correspondence between OCR results in multiple consecutive frames. The agglomeration process selects the most likely character identity in each set of corresponding characters, and therefore the combined result will tend to be more accurate than any of the results from the individual frames.

Specifically, the input to this process is a series of output structures (one for each processed frame) from an OCR engine which produces multiple candidate identities for each recognized text character in the image rank-ordered according to likelihood. A confidence value or probability is associated with the top-ranked character, each of the other candidate characters may also have a confidence value or probability. The output is a series of similar structures, except that a start time (when it first appeared) and an end time (when it disappeared from the image) is associated with each character, word, and line.

The general approach is to successively update an agglomeration ("agg") structure with the OCR results from each video frame as they become available from the OCR engine. The agglomeration process can produce output in different modes of output: 1) output the agglomerated OCR results after every processed frame, 2) output the agglomerated OCR results only when there is a change in the text content of the scene, or 3) output individual text segments from the agglomerated structure immediately after those text segments have disappeared from the video sequence.

Since the segmentation of text lines may vary from frame to frame (because of changes in the scene, or because of slight variations the digitization of the same scene), there may not be a one-to-one correspondence between text lines in successive frames. For example, a line in one frame could overlap two different lines in another frame. In addition, from one frame to the next, the contents of a line of text may change, or characters may be added to a line of text. Therefore, the present approach looks for frame-to-frame changes based on a "character group" structure instead of on a line-by-line basis. A character group is defined as a group of consecutive characters in a text line that each overlap one or more characters in another frame.

A description and definition of the agglomeration algorithm is presented below.

$\{ O_t \}$ is the set of *OCR results* for each frame t. O_t is a hierarchical structure consisting of a set of *lines* $\{ L_i \}$. Each line has a set of *character slots*, and each slot has one or more *candidate character identities*.

$\{ L_i \}$ $i = 1 .. I$ is the set of *lines* of OCR results in the agglomerated ("agg") structure

$\{ L'_i \}$ $i = 1 .. I'$ is the set of *lines* of OCR results from the current ("curr") video frame

{ LP_j } $j= 1 .. J$ is the set of (agg, curr) *line pairs* that overlap
{ T_j } $j= 1 .. J$ is the set of *geometrical transforms* associated with each LP_j

A *character group pair* CGP_k = { { CG_k }, { CG'_k }}, where CG_k are consecutive character slots that overlap one or more character slots in CG'_k , and CG'_k are consecutive character slots that overlap one or more character slots in CG_k . There are five types of correspondences between character slots: 1:1, splits, merges, unaligned, and unmatched.

For each new frame t the algorithm operates as follows:

Step 1. From the two set of lines { L_i} and { L'_i} find pairs of overlapping lines { LP_j} and their associated geometrical transform { T_j}. This step finds the correspondence between text regions in A_t and O_t by one or more methods: a) assuming that the position of the text in the image does not change from frame to frame, test the overlap of bounding boxes of the text lines, b) tracking various pixel data or characteristics of text regions that move from frame to frame (as in Step 220), and c) matching text regions across frames by the identity of their recognized characters, and then computing the geometrical transform between those that match.

Step 2. For each of the line pairs LP_j and its associated geometrical transform T_j find character group pairs CGP_k, $k = 1, \dots K_j$. This step computes character-to-character correspondences by comparing the positions of the left and right sides of the bounding boxes of the character slots. Even when text is stationary in the image, this step permits a small amount of uncertainty in character position due to video capture jitter and differences in character segmentation by the OCR engine.

Step 3. Conditionally update each CGP_k :

3A. If curr text CG'_k is new (different than the corresponding agg text CG_k but in the same image position), and put CG_k on the to-be-replaced list and mark it for output. If both character groups CG'_k and CG_k have a high average confidence and few 1:1 correspondences, or if both character groups have high-confidence characters in 1:1 correspondence that differ radically (i.e., not 0/O, 1/l, etc.), then the text in CG'_k is deemed different than the text in CG_k, and should be replaced.

3B. Else if CG'_k is significantly better (as judged by average confidence value or probability) than CG_k, replace all the characters in CG_k with the characters in CG'_k .

3C. Else update each 1:1, split, merge, and unaligned character slot in CG_k with the corresponding character slot in CG'_k if the latter character slot is significantly better (as judged by average confidence value or probability).

Step 4. Mark text in the agg structure A_t that is not in the curr structure O_t as deletions.

Step 5. Mark text in the curr structure O_t that is not in the agg structure A_t as insertions.

Step 6. Update the agg structure A, with the text marked for replacement, deletion, and insertion. This step updates the characters in CG_k with the characters in CG'_k based on two criteria: the confidence value or probability of each character identity, and frequency count (the number of frames in which each character is present in the OCR results).

Step 7. Output any part of the agg structure A_t that should be output at time t.

As discussed above, the agglomeration process can operate in different output modes: 1) output the entire agglomerated OCR results after every processed frame, 2) output the agglomerated OCR results only when there is a change in the text content of the scene, or 3) output individual text segments from the agglomerated structure immediately after those segments have disappeared from the video sequence. For output modes 1 and 2, Step 6 would be executed before Step 7; for output mode 3, Step 7 would be executed before Step 6.

In an alternate form of agglomeration, the text detection and tracking process 220 forms line sequences; each line sequence corresponds to a single instance of a line of text that appears in multiple images or frames. The OCR results derived from a line sequence are represented by S_t, t = 1, 2, ... T, where T is the # of frames in the sequence. To agglomerate a line sequence, Steps 2 through 6 are run T times; in each run S_t is treated as the "curr" structure. Step 1 is not executed, and Step 7 is executed after agglomerating the last line S_T of the line sequence. This alternate form of agglomeration can also be used to combine multiple OCR results for the same text line appearing in the same frame (obtained after binarizing the same line of text multiple times with different threshold settings, as described in Step 240).

Returning to FIG. 2, in step 270, method 200 applies lexicon processing. Step 270 is achieved by first choosing hypothesized word identities from a lexicon that contain character substrings found in the OCR results produced by step 260. The process then selects the most likely hypothesized words by comparing their characters with the OCR results (including lesser-ranked candidate character identities).

Finally, in step 280, method 200 eliminates false text (i.e., text that is likely to be caused by graphic or other non-text elements in the image). False text is detected by a number of criteria, such as low confidence, brief time appearing in the imagery, and combinations of characters (especially non- alphabetic) within a word that occur infrequently (e.g., "[]Cz-13q").

One important aspect of the present invention is the ability to recognize scene text, such as street signs, name plates, and billboards, that is part of the video scene itself. Many text recognition efforts in video and still imagery have assumed that the text lies in a plane that is oriented roughly perpendicular to the optical axis of the camera. Of course, this assumption is valid for scanned document images and imagery containing overlaid text captions, but is not generally true for scene text. In fact, text information is often viewed from an oblique angle. Such a configuration is quite common when the main subject of the scene is not the text itself, but such incidental text could be quiet important (for example, it may be the only clue of the location of the captured imagery).

To address the problem of recognizing text that lies on a planar surface in 3-D space, one should note that the orientation angle of such text relative to the camera can be modeled in terms of three angles, as shown in FIG. 3:

- θ, the rotation in the plane perpendicular to the camera's optical axis
- φ and γ, the horizontal (azimuth) and vertical (elevation) components, respectively, of the angles formed by the normal to the text plane and the optical axis.

The three angles represent the amount of rotation that the text plane must undergo relative to the camera in each of its three axes to yield a frontal, horizontal view of the plane in the camera's field of view. When θ and γ are zero and φ is nonzero, the apparent width of the text is reduced, resulting in a change in aspect ratio and a loss of horizontal resolution. Similarly, when θ and φ are zero and γ is nonzero, the text appears to be squashed vertically. The severity of perspective distortion is proportional to D/Z, where D is the extent of the text parallel to the optical axis (its "depth") and Z is the distance from the text to the camera. When the text is not centered at the optical axis or both φ and γ are nonzero, the text appears to be rotated in the image plane (see FIG. 4). If the text were rotated to remove this apparent angle by a text recognition process that mistakenly assumed the text is fronto-parallel, the characters would become sheared (see FIG. 5). When both φ and γ are nonzero and perspective distortion is significant, the shearing angle varies from left to right within the text region. OCR engines perform poorly if the shearing causes characters to touch or to be severely kerned (overlapped vertically).

When the plane that contains the text is at an angle relative to the image plane, several types of distortions can be introduced that make it difficult to read the text. In the most general case, the distortion is described as a projective transformation (or homography) between the plane containing the text and the image plane. The present invention can correct this distortion by applying the appropriate "corrective" projective transformation to the image. That is, the present method can rotate and stretch the original image to create a synthetic image, which is referred to as a "rectified image," in which the projective distortion has been removed.

In general, a two-dimensional projective transformation has eight degrees of freedom. Four correspond to a Euclidean 2-D transformation (translations along two axes t_x and t_y, a rotation r, and an isotropic scale factor s); two correspond to an affine transformation (a shear a and a nonisotropic scaling b of one axis relative to the other); and the remaining two degrees of freedom represent a perspective foreshortening along the two axes f_x and f_y.

From an OCR point of view, some of the eight parameters produce changes that are harder to handle than others. In particular, the two translations are not a problem, because they simply produce an image shift that is naturally handled by OCR systems. Similarly, the two scale factors are not a problem, because the OCR systems typically include mechanisms to work at multiple scales. The Euclidean rotation is important, but is easily computed from a line of text. Therefore, three critical parameters produce distortions that are difficult for OCR systems to handle: the two perspective foreshortening parameters and the shearing.

In the present approach, estimates of the plane parameters are computed from the orientations of the lines of text in the image and the borders of planar patch, if they are visible. To remove a projective distortion, the present invention computes the three critical degrees of freedom associated with the plane on which the text is written. In general, the present invention can accomplish this task by identifying three geometric constraints associated with the plane. For example, one can compute the necessary parameters, given two orthogonal pairs of parallel lines, such as the borders of a rectangular sign or two parallel lines of text and a set of vertical strokes within the text. The three constraints derivable from these sets of lines are two vanishing points (one from each set of parallel lines) and an orthogonality constraint between the set of lines.

Sometimes, however, such linear properties are difficult to detect. In such cases, the present invention can estimate the parameters by making assumptions about the camera-to-plane imaging geometry that are often true. For example, people normally take pictures so that the horizon is horizontal in the image. In other words, they seldom rotate the camera about its principal axis. In addition, they often keep the axis of the camera relatively horizontal. That is, they do not tilt the camera up or down very much. When these two assumptions apply and the text lies on a vertical plane, such as a wall of a building or a billboard, the projective distortion is only along the X axis of the image. The perspective foreshortening in that direction can be computed from one constraint, such as a pair of horizontal parallel lines.

Another assumption that often holds is that the perspective effects are significantly smaller than the effects caused by the out-of-plane rotations. This is the case if the depth variation in the text is small compared with the distance from the camera to the plane. In this case, the perspective distortion is reduced to an affine shear and the projection is described as a weak perspective projection.

Given these relationships, the general strategy is to identify as many properties of a region of text as possible, and then compute a corrective transformation, using as few assumptions as possible. Initially, the present invention uses information derived independently from each individual line of text. Next, the present invention combines information from multiple text lines after partitioning them into sets of lines that lie within a common plane. The method then further augments the process by detecting auxiliary lines that can provide horizontal and vertical cues. These can include lines in the same plane as the text (such as sign borders), and extraneous lines (e.g., building edges). Finally, depending upon the success in finding these features, one can either make assumptions to substitute for missing constraints (and then compute a transformation that corrects for a full perspective projection) or compute a transformation that does not completely remove all degrees of freedom.

In one embodiment, each text line is rectified in a single image independently. After possible lines of text are detected, various features of each text line are then estimated. These include the top and base lines, and the dominant vertical direction of the character strokes. The rectification parameters for each text line are computed from these characteristics. Each text line is then rectified independently and sent to an OCR engine.

Figure 6 illustrates a test image of a poster containing text that was captured at an azimuth angle of 70 degrees; the rectangles that have been fitted to each detected text line are shown in overlay. (Some of the rectangles do not look to the eye like true rectangles because of the perspective view of the image contents). Computing the best-fitting rectangle for each text line is an expedient way to approximate the location and extent of the text, but the top and bottom of the text are not accurately computed when significant perspective distortion is present.

A top line and base line for each line of text are estimated by rotating the text line at various angles and then computing a series of horizontal projections 720 over the vertical edge transitions 710 in FIG. 7. (When the text consists of predominantly lower-case characters, the “top” line actually corresponds to the “midline” of the text that touches the tops of lower-case characters, excluding their ascenders.) The best estimate of the bottom line should correspond to the rotation angle that yields the steepest slope on the bottom side of the horizontal projection. Specifically, to refine the bottom baseline, a bottom edge pixel is located for each column in the initial rectangular bounding box. These bottom edge pixels are then rotated through a series of angles around the original estimated text angle and summed horizontally along each row, P[r]. The angle with the maximum sum of squared projections (sum(P[r]*P[r])) is the baseline angle and the maximum projection is the baseline position. For speed, a binary search is used to locate this maximum projected angle. The process is repeated for the top baseline using only the top edge pixels in each column. FIG. 7 shows an example of this procedure.

It should be noted that in addition to computing two horizontally oriented lines, one would like to find and measure the angles of two vertically oriented lines to use in the computation of the rectification parameters. Unfortunately, an individual line of text does not have much vertical extent, and it is difficult to determine which parts of the text could be used as vertical cues. However, the height of the text is not usually a significant fraction of the depth of the text in 3-D space, so that the perspective foreshortening in the Y dimension should be relatively small. Therefore, in the absence of any other reliable vertical cues, the present method computes the dominant vertical direction (shear) of the text by computing a series of vertical projections 820 over the vertical edge transitions after rotation the text line in 2-degree increments. The best estimate of the dominant vertical direction should correspond to the angle at which the sum of squares of the vertical projection is a maximum (on the assumption that the projection of true vertical strokes is greatest when they are rotated to a vertical position). FIG. 8 shows an example of shear computation.

FIG. 9 illustrates the refined bounding boxes based on the top and base lines and on the dominant vertical direction. FIG. 10 illustrates the rectified text lines (a) after the initial rectangular bounding box is deskewed; (b) after the baseline is refined (without including the top line in the dewarping computation) and then deskewed; and (c) after the lines are desheared.

The transformation used to rectify the image of each text line, $\mathbf{I}_j$, occurring in an obliquely viewed image, $\mathbf{O}_i$, is a projective transformation, $\mathbf{T}_{ij}$, of the text plane. This transformation is described by:

$$m' = Hm \quad ,$$

where H is a 3 x 3 matrix that maps the homogeneous coordinates $m = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$ in $\mathbf{O}_i$ to the homogeneous rectified coordinates m′ = EAPm in a normalized image $\mathbf{N}_i$, where:

m= coordinates in the video image
m'= coordinates in the normalized image
E= Euclidian transformation
A= Affine transformation
P= Projective transformation

where:

$$E = \begin{pmatrix} s\cos \mathrm{r} & s\sin \mathrm{r} & t_x \\ s\sin \mathrm{r} & s\cos \mathrm{r} & t_y \\ 0 & 0 & 1 \end{pmatrix} \quad A = \begin{pmatrix} 1/\mathrm{b} & -\mathrm{a}/\mathrm{b} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad P = \begin{pmatrix} \mathrm{f} & 0 & 0 \\ 0 & \mathrm{f} & 0 \\ l_x & l_y & \mathrm{f} \end{pmatrix}$$

The horizontal and vertical vanishing points are mapped to the points at infinity in the horizontal and vertical directions. This process takes care of the perspective foreshortening in both directions, as well as the skew and rotation. The remaining four degrees of freedom correspond to the origin and scale factors that place the line in the normalized image $\mathbf{N}_i$. The image $\mathbf{N}_i$, which contains all of the rectified lines from image $\mathbf{O}_i$, is then sent through the OCR process. FIG. 11 shows, for the 70 degree azimuth test image, with the recognition results overlaid on the normalized image.

FIG. 12 illustrates a block diagram of a portable text recognition device or system 1200 of the present invention. In one embodiment, the portable text recognition device or system 1200 is implemented using a general purpose computer or any other hardware equivalents. More specifically, the recognition device or system 1200 is preferably implemented as a portable device. In an alternative embodiment, all or various components of system 1200 can be adapted to a digital video camera or digital still camera.

Thus, text recognition device or system 1200 comprises a processor (CPU) 1230, a memory 1240, e.g., random access memory (RAM) and/or read only memory (ROM), a text recognition and extraction engine 1220, and various input/output devices 1210, (e.g., storage devices 111, including but not limited to, a tape drive, a floppy drive, a hard disk drive or a compact disk drive), a receiver 1212, a transmitter 1213, a speaker 1214, a display 1215, an image capturing sensor 1216, e.g., those used in a digital still camera or digital video camera, a clock 1217, an output port 1218, a user input device 1219 (such as a keyboard, a keypad, a mouse, and the like, or a microphone for capturing speech commands).

It should be understood that the text recognition and extraction engine 1220 can be implemented as physical devices that are coupled to the CPU 1230 through a communication channel. Alternatively, the text recognition and extraction engine 1220 can be represented by one or more software applications (or even a combination of software and hardware, e.g., using application specific integrated circuits (ASIC)), where the software is loaded from a storage medium, (e.g., a magnetic or optical drive or diskette) and operated by the CPU in the memory 1240 of the computer. As such, the text recognition and extraction engine 1220 (including associated data structures) of the present invention can be stored on a computer readable medium, e.g., RAM memory, magnetic or optical drive or diskette and the like.

The text recognition and extraction engine 1220 comprises a text detection module 1221, a text orientation module 1222, a text binarization module 1223, an optical character recognition (OCR) module 1224, an agglomeration module 1225, a lexicon processing module 1226, and a false text detection module 1227. In operation, the text recognition and extraction engine 1220 is able to accurately detect and extract text information from an input image or video imagery. A detailed description of the functions of the text recognition and extraction engine 1220 is disclosed below with reference to FIG. 13. The text results from the text recognition and extraction engine 1220 is then provided to the processor 1230 and application software module 1250 for further processing to provide various functionalities or services. The application software module 1250 implements these functionalities or services, which include, but are not limited to, automatic text recognition with audio playback or visual display (e.g., street signs, placards, restaurant menus, billboards, white boards, labels, or books), automatic text translation to a foreign language, automatic access to auxiliary information, automatic road sign reading for navigation, automatic license plate reading for law enforcement functions, image and video indexing and archiving and inventory and shelf restocking control. Each of these embodiments is further discussed below.

It should be noted that seven (7) different embodiments of the present invention are described below. Since each embodiment provides different functionality, the hardware and software requirements are different for each of the embodiments. As such, the text recognition device or system 1200 of FIG. 12 is illustrated with various elements in solid lines and dash lines. The elements in solid lines are those elements that are typically considered as required elements, whereas elements in dashed lines are considered optional elements. Thus, although FIG. 12 serves as a block diagram for all seven embodiments as described below, it should be understood that each embodiment may comprise all or only a subset of all the elements as shown in FIG. 12.

FIG. 13 illustrates a method of utilizing the portable text recognition system of the present invention in a first embodiment. In a first embodiment, the present device is deployed as a portable language translator. For example, a user travelling in a foreign country can capture an imagery having text (e.g., taking a picture of a restaurant menu, a transit schedule, signs, placards). The text within the captured imagery is detected and translated to a native language of the user. A pertinent language translator can be loaded into the portable device.

Specifically, the method is designed for portably detecting and reading text appearing in video or still imagery. The system 1200 of FIG. 12 employing method 1300 can accept a video or still image signal and recognize text in real time. It should be noted that the term "captured imagery" in the present application may encompass, in part or in whole, a single still image or video frame, and/or a sequence of still images or video frames.

Method 1300 starts in step 1305 and proceeds to step 1310, where an image or an image sequence (e.g., video) is captured via conventional equipment, e.g., image capturing sensor 1216. Alternatively, step 1310 can be omitted if the captured imagery was previously captured and is simply being retrieved from a storage device 1211.

In step 1320, method 1300 detects, locates, and tracks text region within the captured imagery. Different text region detection, location, and tracking methods can be employed in step 1320. For example, a text detection method is disclosed in US patent application entitled "Method And Apparatus For Recognizing Text In An Image Sequence Of Scene Imagery" with attorney docket SRI/4483-2, which is herein incorporated by reference and is filed simultaneous herewith.

In brief, method 1300 approaches text detection and location with the assumption that the text is roughly horizontal, and that the characters have a minimum contrast level compared with the image background. The text may be of either polarity (light text on a dark background, or dark text on a light background). The method first detects vertically oriented edge transitions in the gray-scale image, using a local neighborhood edge operator. The output of the operator is thresholded to form two binary images, one for dark-to-light transitions (B1), and the other for light-to-dark transitions (B2). A connected components algorithm is applied on each binary image. The connected components that have been determined (by examining their height and area) not due to text are eliminated. The remaining connected components are linked to form lines of text by searching the areas to the left and right of each connected component for additional connected components that are compatible in size and relative position. Finally, a rectangle is fitted to each line of detected text or a group of lines of text using the moments of all connected components used to locate the text. Tracking text over multiple consecutive video frames Is achieved by computing the frame-to-frame displacement of the pixels in a set of local neighborhoods, or finding and following distinctive shape features, such as the ends of character strokes, and then computing a geometric transformation that characterizes the frame-to-frame displacement of corresponding text regions.

In step 1330, method 1300 may optionally adjust the detected text to account for orientation. Namely, text in a captured imagery is often viewed from an oblique angle. Such a configuration is quite common when the main subject of the scene is not the text itself, but such incidental text could be quiet important (for example, it may be the only clue of the location of the captured imagery). Thus, method 1300 may apply a processing step in step 1330 to account for text orientation, thereby improving the OCR method that will be applied at a later processing stage. Example of an orientation adjustment method of step 1330 is again provided in US patent application with attorney docket SRI/4483-2, which is filed simultaneous herewith.

In step 1340, method 1300 optionally applies binarization of the detected text regions. Binarization is performed on each text line independently. If the OCR processing 1350 operates on a gray-scale imagery instead of on binary images, the processing in step 1340 would not be required. Different binarization methods can be employed in step 1340. For example, a binarization method is disclosed in US patent application with attorney docket SRI/4483-2.

In brief, step 1340 performs binarization on each text line by first determining the polarity of the text line, and then performing binarization of the text line. The polarity is determined by comparing grayscale pixel values above and below the baselines. This relies on the inside pixels (those below the top and above the bottom baselines) most likely being character pixels and the outside pixels (those above the top and below the bottom baseline) most likely being background pixels. The polarity calculation compares pairs of pixels along both baselines and sums the number of times the inside pixel is greater than the outside pixel. If this sum is greater than zero, the polarity is determined to be light text on a dark background; otherwise, the polarity is determined to be dark text on a light background.

In binarization, the gray-scale image is smoothed with a Gaussian kernel, and histograms H1 and H2 are computed. Histogram H1 is composed of gray-scale pixels in the smoothed image on the right side of the connected components in the dark-to-light edge transition image B1 and on the left side of the light-to-dark edge transition image B2. If light text is in this text region, these are the pixels most likely to belong to light text or near the edge of light text. Similarly, histogram H2 is composed of gray-scale pixels in the smoothed image on the right side of the connected components in image B2 and on the left side of the image B1. The threshold for the text line is then set to the gray value at the 60th percentile of histogram H1 or H2, depending on the polarity chosen. Alternatively, more than one binarizaton result for each text line is produced, each using a different threshold value (e.g., 45th percentile, 60th percentile, and 75th percentile). Producing more than one binarization result, and sending them through the OCR process (Step 1350) can, after combining the OCR results with agglomeration (Step 1360), sometimes yield more accurate results than processing a single binarization result.

Returning to FIG. 12, in step 1350, method 1300 applies OCR processing to the text regions. In one embodiment, step 1350 is achieved by using a commercially available OCR engine e.g., an OCR package from Scansoft, Inc. of Peabody, MA. However, it should be noted the present invention is not so limited and that other OCR packages may also be used. It should be noted that some OCR engines operate on a gray-scale imagery instead of binary images and therefore would not require the processing in step 1340. The OCR engine produces one or more candidate identities for each recognized text character in the image, rank-ordered according to likelihood.

In step 1360, method 1300 may optionally agglomerate the OCR results. Specifically, a video text recognition process usually involves performing optical character recognition (OCR) on images derived from individual video frames. However, in many applications the same text persists in the scene for some length of time. Digitized video frames of the same scene may vary slightly, thereby causing an OCR process operating on individual frames to produce slightly different results. Therefore, method 1300 may combine ("agglomerate") OCR results from multiple frames, in a manner that takes the best recognition results from each frame and forms a single result. The use of agglomeration improves the recognition accuracy over that of the OCR results on individual images. It also enables the system to avoid outputting the same results repeatedly when the text is persistent in the video sequence for many frames, and reduces the generation of false characters from non-text image regions. In addition, because the agglomeration process works on OCR results (as opposed to image pixels) from multiple frames, it is computationally fast enough to implement in a real-time system (i.e. one that keeps up with the video display rate). Example of an agglomeration method is disclosed in US patent application with attorney docket SRI/4483-2.

In step 1370, method 1300 may optionally apply lexicon processing. Step 1370 is achieved by first choosing hypothesized word identities from a lexicon that contain character substrings found in the OCR results produced by step 1360. The process then selects the most likely hypothesized words by comparing their characters with the OCR results (including lesser-ranked candidate character identities). The contents of the lexicon are dynamically determined based on the information context – for example, by the task (such as a list of breakfast cereals for grocery shopping), or by the location or environment that the user is operating in (such as a geographic gazetteer for navigation). The contents of the lexicon may be selected from files pre-loaded on the Portable Text Recognition Device 1200, or it may be accessed from the web via a wireless link via receiver 1212 and transmitter 1213 during operation of the device.

In step 1380, method 1300 may optionally eliminate false text detection (e.g., low confidence and non-alphabetic text). Specifically, method 1300 looks for OCR results containing low-confidence and non-alphabetic text that are likely to be caused by graphic or other non-text elements in the image. Example of a false text detection method of step 1380 is again provided in US patent application with attorney docket SRI/4483-2, which is filed simultaneous herewith.

In step 1382, method 1300 may optionally correlate supplemental information in accordance with the detected text information. For example, if the user is travelling in Germany and has taken a picture of a menu in German, then method 1300 may optionally provide information relating to certain detected words in the menu. For example, white asparagus is a seasonal produce and is strongly favored by Germans during the late spring season. If the term for white asparagus is detected, method 1300 in step 1382 may correlate this detected term with additional information that is retrieved for the user. This optional step can be employed in conjunction with step 1370 where a lexicon pertaining to travel to Germany is previously loaded in a storage 1211 of the portable text recognition device 1200. Alternatively, if receiver 1212 and transmitter 1213 are deployed, then the correlated supplemental information can be retrieved and downloaded into the portable text recognition device 1200.

Another example is where the user is travelling in a foreign country and has captured an imagery that contains a street sign. Method 1300 may then optionally provide supplemental information relating to the detected street name. For example, method 1300 may provide a list of restaurants, hotels, metro stations, bus stops, and famous landmarks that are in the immediate vicinity to the user. It should be noted that the term “travel information” as used in the present application comprises one or more of the following information: restaurants, hotels, train stations, bus stops, airports, landmarks, emergency facilities (e.g., police stations and fire stations) and street names and numbers.

In yet another example, the recognized text could also be used as landmarks that help locate where the user is relative to a map, in what direction the user is looking, and what the user is looking at. In fact, a local map can be retrieved from a storage device 1211 to show the current location to the user. Thus, portable text recognition device 1200 can be implemented as a portable travel assistant, thereby providing navigational help through complex or unfamiliar surroundings, such as for a tourist in a foreign city environment.

In step 1384, method 1300 applies language translation. Namely, the detected text information is sent to a language translation module stored in storage device 1211 to convert the recognized text into the user’s native language. It should be noted that steps 1382 and 1384 are implemented in the application software module 1250.

In step 1386, method 1300 outputs the result visually and/or audibly to the user. Specifically, the result can be provided to the user via a display (e.g., LCD display) and/or a text-to-speech synthesis process and the speaker 1214. It should be noted that the result can also be stored in a storage device 111 for later retrieval. In an alternative way to implement this embodiment, the detected text regions generated by step 1320 could be indicated or highlighted on the display 1215, thus allowing the user to select via a user input device 1219 which text regions should be recognized and translated. Method 1300 then ends in step 1390.

FIG. 14 illustrates a method of utilizing the portable text recognition system of the present invention in a second embodiment. In this second embodiment, the present device is deployed as a portable assistant to an individual who is visually impaired or who needs reading assistance. For example, a user shopping in a store can capture an imagery having text (e.g., taking a picture of the label of a product). Another example is a child taking a picture of a page in a book. The text within the captured imagery is detected and audibly broadcasted to the user via a speaker.

Thus, the portable text recognition device 1200 can help a sight-impaired person navigate in an urban or commercial environment, select products from a grocery store shelf, read the label on a prescription bottle, or operate a vending machine. The recognized text would be sent to a speech synthesis module 1252 stored in a storage device that produces an audio form via speaker 1214 for the person with impaired sight to hear. Thus, portable text recognition device 1200 can be a portable book reader for the sight impaired, or for children.

Specifically, method 1400 starts in step 1405 and proceeds to step 1410. It should be noted that steps 1410-1480 are similar to steps 1310-1380. As such, the description for steps 1410-1480 is provided above.

In step 1482, method 1400 may optionally apply language translation if the detected text is not in the native language of the user. An example is where the visually impaired user is traveling abroad or the user is reading a book in a foreign language. It should be noted that step 1482 is implemented in the application software module 1250.

In step 1484, method 1400 outputs the result audibly to the user via a speaker. However, the result can also be provided to the user via a display (e.g., LCD display). It should be noted that the result can also be stored in a storage device 1211 for later retrieval. Method 1400 then ends in step 1490.

FIG. 15 illustrates a method of utilizing the portable text recognition system of the present invention in a third embodiment. In this third embodiment, the present device is deployed as a portable notebook. For example, a user in an educational environment can capture an imagery having text (e.g., taking a picture of a white board, view graph or a screen). The text within the captured imagery is detected and stored in a format that can be retrieved later for text processing, e.g., in a word processor format.

Specifically, method 1500 starts in step 1505 and proceeds to step 1510. It should be noted that steps 1510-1580 are similar to steps 1310-1380. As such, the description for steps 1510-1580 is provided above.

In step 1582, method 1500 may optionally apply language translation if the detected text is not in the native language of the user. An example is where a user is attending a seminar, a class or a meeting where a foreign language is used. Again, this optional step can be employed in conjunction with step 1570 where a lexicon pertaining to education topics (e.g., with specific technical terms pertaining to a specific field) can be previously loaded in a storage 1211 of the portable text recognition device 1200. It should be noted that step 1582 is implemented in the application software module 1250.

In step 1584, method 1500 outputs the result visibly to the user via a display (e.g., LCD display). It should be noted that the result can also be stored in a storage device 1211 for later retrieval, e.g., as a word processing file. Method 1500 then ends in step 1590.

FIG. 16 illustrates a method of utilizing the portable text recognition system of the present invention in a fourth embodiment. In this fourth embodiment, the present device is deployed as a portable auxiliary information accessor. For example, a user in a business environment can capture an imagery having text (e.g., taking a picture of a bill board or a business card having an Internet or web address). The text within the captured imagery is detected and the Internet address is accessed to acquire additional information.

For example, a billboard ad may have a web address that contains more information about the product (perhaps even an audio or video clip) that could be immediately retrieved. The web address can be accessed via transmitter 1213 and receiver 1212.

Another example is where a user may receive a business card at a trade show and be able to immediately retrieve information from that person's home page, or a softcopy version of a printed document can be retrieved. The user can communicate with other remote people about the document rather than faxing the document or reading off the web address of the document, or get additional product information off the web, such as competitive pricing or product reliability.

Specifically, method 1600 starts in step 1605 and proceeds to step 1610. It should be noted that steps 1610-1680 are similar to steps 1310-1380. As such, the description for steps 1610-1680 is provided above.

In step 1682, method 1600 correlates supplemental information based upon the detected text, e.g., a web address. The supplemental information is retrieved via the receiver 1212 and transmitter 1213. It should be noted that step 1682 is implemented in the application software module 1250.

In step 1684, method 1600 outputs the result visibly to the user via a display (e.g., LCD display). It should be noted that the result can also be stored in a storage device 1211 for later retrieval, e.g., as a word processing file. Method 1600 then ends in step 1690.

FIG. 17 illustrates a method of utilizing the portable text recognition system of the present invention in a fifth embodiment. In this fifth embodiment, the present device is deployed as a portable navigation assistant. For example, the portable unit is deployed in a vehicle for automatic reading of road signs and speed limit signs. The text within the captured imagery is detected and is provided to the computer in the vehicle for assisting the vehicle's navigation system or as a warning indicator to the driver on an instrument panel for speed limit monitoring.

Specifically, method 1700 starts in step 1705 and proceeds to step 1710. It should be noted that steps 1710-1780 are similar to steps 1310-1380. As such, the description for steps 1710-1780 is provided above.

In step 1782, method 1700 correlates supplemental information based upon the detected text, e.g., road signs, highway numbers, exit numbers and the like. For example, method 1700 may provide a list of restaurants, hotels, and famous landmarks that are in the immediate vicinity to the user based upon the road signs, highway numbers, and/or exit numbers. It should be noted that step 1782 is implemented in the application software module 1250.

In step 1784, method 1700 outputs the result visibly or audibly to the user via a display (e.g., LCD display) or a speaker and directly to the vehicle's navigational system via an output port 1218. It should be noted that the result can also be stored in a storage device 1211 for later retrieval.

For example, the portable text recognition system 1200 may simply maintain a history log of detected road signs and exit numbers. Thus, if the vehicle breaks down on a highway and the driver is unable to recall which exit or roadway the vehicle is closest to, the driver can simply retrieve the history log to see which exit or roadway that the driver has recently encountered. The clock 1218 can also be utilized to time stamp each occurrence of detected text, thereby allowing the driver to accurately communicate the location of his stranded vehicle and the approximate time from a text detection event, e.g., 5 minutes from exit 5 and so on.

FIG. 18 illustrates a method of utilizing the portable text recognition system of the present invention in a sixth embodiment. In this sixth embodiment, the present device is deployed as a portable law enforcement assistant. For example, the portable unit is deployed in a police vehicle for reading license plates, vehicle identification numbers (VINs) or driver licenses and registrations. The text within the captured imagery is detected and is used to provide information to a law enforcement officer as to the status of a vehicle or a driver. It should be noted that the term "vehicle information" as used in the present application comprises one or more of the following information: license plate numbers, vehicle identification numbers (VINs), driver license numbers, registration numbers, current status of license holder's driving privilege, status of vehicle (e.g., currently registered, not registered, reported as stolen and so on). In addition, vehicle information includes boats registration numbers.

Examples may include but not limited to an attachment to a police radar gun, felon detection by reading and running license plates autonomously, and stolen vehicle identification, parking lot access, billing and vehicle security. Namely, the police officer can automatically enter vehicle license plate information as the officer walks or drives down a city street for timed parking violations (e.g., via time stamp with clock 1217), or automatically entering driver's license ID information after a person has been stopped by the police.

Specifically, method 1800 starts in step 1805 and proceeds to step 1810. It should be noted that steps 1810-1880 are similar to steps 1310-1380. As such, the description for steps 1810-1880 is provided above.

In step 1882, method 1800 correlates supplemental information based upon the detected text, e.g., a plate number or a driver license. The supplemental information is retrieved via the receiver 1212 and transmitter 1213. It should be noted that step 1882 is implemented in the application software module 1250.

In step 1884, method 1800 outputs the result visibly or audibly to the user via a display (e.g., LCD display) or a speaker and directly to the officer's motor vehicle database system via an output port 1218. It should be noted that the result can also be stored in a storage device 1211 for later retrieval. Method 1800 then ends in step 1890.

FIG. 19 illustrates a method of utilizing the portable text recognition system of the present invention in a seventh embodiment. In this seventh embodiment, the present device is deployed as a portable inventory assistant. For example, a user in a store or a warehouse can capture an imagery having text (e.g., taking a picture of a product on a shelf or high up on a scaffold). The text within the captured imagery is detected and is used for inventory control. Namely, the portable text recognition device 1200 can control inventory and shelf restocking (as an alternative identification technology to bar code reading). In another example, the odometer reading for a returned rental car could be automatically captured.

Specifically, method 1900 starts in step 1905 and proceeds to step 1910. It should be noted that steps 1910-1980 are similar to steps 1310-1380. As such, the description for steps 1910-1980 is provided above.

In step 1982, method 1900 may optionally correlate supplemental information based upon the detected text, e.g., brand name and generic product name. The supplemental information may include but is not limited to the current volume of a particular product in stock, the status as to shipment of a particular product, the cost of a particular product in stock, and the like. The supplemental information is retrieved via the receiver 1212 and transmitter 1213. It should be noted that step 1982 is implemented in the application software module 1250.

In step 1984, method 1900 outputs the result visibly or audibly to the user via a display (e.g., LCD display) or a speaker. It should be noted that the result can also be stored in a storage device 1211 for later retrieval. Method 1900 then ends in step 1990.

Finally, the portable text recognition device 1200 can also index and archive image and video, both for storage identification, and as a means to increase the accuracy of targeted marketing programs. An example of this is to apply this technique on an internet photo server using the results to increase the accuracy that the pop up ads the user seeks is relevant.

Thus, the portable text recognition device 1200 can be implemented to provide different levels of functionality with different hardware and software complexity. Although each embodiment can be implemented and manufactured as a dedicated unit for a particular application, the portable text recognition device 1200 can be designed to receive upgrade modules (in hardware form or software form) to implement one or more of the above disclosed embodiments.

Although various embodiments which incorporate the teachings of the present invention have been shown and described in detail herein, those skilled in the art can readily devise many other varied embodiments that still incorporate these teachings.

What is claimed is:

1. Method for recognizing text in a captured imagery, said method comprising the steps of:

(a) detecting a text region in the captured imagery;

(b) adjusting said detected text region to produce a rectified image; and

(c) applying optical character recognition (OCR) processing to said rectified image to recognize the text in the captured imagery.

2. The method of claim 1, wherein said adjusting step (b) comprises the step of (b1) computing a base line and a top line for a line of detected text within said detected text region.

3. The method of claim 2, wherein said base line and said top line correlate substantially to horizontal parallel lines of a rectangular bounding box that is fitted to said line of detected text.

4. The method of claim 2, wherein said base line and said top line are estimated by rotating said line of detected text at various angles and then computing a plurality of horizontal projections over a plurality of vertical edge projections.

5. The method of claim 4, wherein said base line is selected that corresponds to a rotation angle that yields a steepest slope on a bottom side of one of said plurality of horizontal projections.

6. The method of claim 4, wherein said top line is selected that corresponds to a rotation angle that yields a steepest slope on a top side of one of said plurality of horizontal projections.

7. The method of claim 2, wherein said base line is selected comprising the steps of:

locating a plurality of bottom edge pixels, where each bottom edge pixel is located for each column in said rectangular bounding box;

rotating said plurality of bottom edge pixels through a series of angles around an initial estimated text angle for said line of detected text;

summing horizontally along each row; and

determining a baseline angle from a maximum sum of squared projections and determining a baseline position from a maximum projection.

8. The method of claim 2, wherein said top line is selected comprising the steps of:

locating a plurality of top edge pixels, where each top edge pixel is located for each column in said rectangular bounding box;

rotating said plurality of top edge pixels through a series of angles around an initial estimated text angle for said line of detected text;

summing horizontally along each row; and

determining a top line angle from a maximum sum of squared projections and determining a top line position from a maximum projection.

9. The method of claim 2, wherein said adjusting step (b) further comprises the step of (b2) computing a dominant vertical direction of character strokes for a line of detected text within said detected text region.

10. The method of claim 9, wherein said dominant vertical direction computing step (b2) comprises the step of computing a plurality of vertical projections over a plurality of vertical edge transitions after rotating said line of detected text in a plurality of degree increments.

11. The method of claim 10, wherein said dominant vertical direction is selected that corresponds to an angle where a sum of squares of said vertical projections is a maximum.

12. The method of claim 1, further comprising the step of:

(b1) binarizing said detected text region prior to applying said OCR processing step (c).

13. The method of claim 12, further comprising the step of:

(d) applying agglomeration processing subsequent to said OCR processing to produce the text in the captured imagery.

14. The method of claim 13, further comprising the step of:

(e) applying lexicon processing subsequent to said agglomeration processing to produce the text in the captured imagery.

15. The method of claim 14, further comprising the step of:

(f) applying false text elimination processing subsequent to said lexicon processing to produce the text in the captured imagery.

16. Apparatus (100) for recognizing text in a captured imagery, said apparatus comprising:

means (121) for detecting a text region in the captured imagery;

means (122) for adjusting said detected text region to produce a rectified image; and

means (124) for applying optical character recognition (OCR) processing to said rectified image to recognize the text in the captured imagery.

17. The apparatus of claim 16, wherein said adjusting means computes a base line and a top line for a line of detected text within said detected text region.

18. The apparatus of claim 17, wherein said base line and said top line correlate substantially to horizontal parallel lines of a rectangular bounding box that is fitted to said line of detected text.

19. The apparatus of claim 17, wherein said base line and said top line are estimated by rotating said line of detected text at various angles and then computing a plurality of horizontal projections over a plurality of vertical edge projections.

20. The apparatus of claim 19, wherein said base line is selected that corresponds to a rotation angle that yields a steepest slope on a bottom side of one of said plurality of horizontal projections.

21. The apparatus of claim 19, wherein said top line is selected that corresponds to a rotation angle that yields a steepest slope on a top side of one of said plurality of horizontal projections.

22. The apparatus of claim 17, wherein said adjusting means further computes a dominant vertical direction of character strokes for a line of detected text within said detected text region.

23. The apparatus of claim 22, wherein said adjusting means computes said dominant vertical direction by computing a plurality of vertical projections over a plurality of vertical edge transitions after rotating said line of detected text in a plurality of degree increments.

24. Method for recognizing text in a captured imagery having a plurality of frames, said method comprising the steps of:

(a) detecting a text region in a frame of the captured imagery;

(b) applying optical character recognition processing (OCR) to said detected text region to identify potential text for said frame; and

(c) agglomerating the OCR identified potential text over a plurality of frames in the captured imagery to recognize the text in the detected text region.

25. The method of claim 24, wherein said agglomerating step (c) comprises the step of updating an agglomeration structure with said OCR identified potential text of a current frame.

26. The method of claim 25, wherein said updating step comprises the step of (c1) finding correspondence between a text region of said agglomeration structure with a text region of said current frame.

27. The method of claim 26, wherein said updating step further comprises the step of (c2) finding character-to-character correspondence for each pair of overlapping lines between said text region of said agglomeration structure with said text region of said current frame to find one or more character group pairs.

28. The method of claim 27, wherein said updating step further comprises the step of (c3) updating said one or more character group pairs.

29. The method of claim 28, wherein said updating step further comprises the step of (c4) marking text in said agglomeration structure that is not in said current frame as a deletion.

30. The method of claim 29, wherein said updating step further comprises the step of (c5) marking text in said current frame that is not in said agglomeration structure as an insertion.

31. The method of claim 25, further comprising the step of:

(d) outputting said text in the detected text region after each processed frame.

32. The method of claim 25, further comprising the step of:

(d) outputting said text in the detected text region only when a change is detected as to said text of said captured imagery.

33. The method of claim 25, further comprising the step of:

(d) outputting only said text within said agglomeration structure when said text is not detected in a current frame.

34. Apparatus (100) for recognizing text in a captured imagery having a plurality of frames, said apparatus comprising:

means (121) for detecting a text region in a frame of the captured imagery;

means (124) for applying optical character recognition processing (OCR) to said detected text region to identify potential text for said frame; and

means (125) for agglomerating the OCR identified potential text over a plurality of frames in the captured imagery to extract the text in the detected text region.

35. The apparatus of claim 34, wherein said agglomerating means updates an agglomeration structure with said OCR identified potential text of a current frame.

36. The apparatus of claim 35, wherein said agglomerating means finds correspondence between a text region of said agglomeration structure with a text region of said current frame.

37. The apparatus of claim 36, wherein said agglomerating means further finds character-to-character correspondence for each pair of overlapping lines between said text region of said agglomeration structure with said text region of said current frame to find one or more character group pairs.

38. The apparatus of claim 37, wherein said agglomerating means further updates said one or more character group pairs.

39. The apparatus of claim 38, wherein said agglomerating means further marks text in said agglomeration structure that is not in said current frame as a deletion.

40. The apparatus of claim 39, wherein said agglomerating means further marks text in said current frame that is not in said agglomeration structure as an insertion.

41. The apparatus of claim 35, further comprising:

means (110) for outputting said text in the detected text region after each processed frame.

42. The apparatus of claim 35, further comprising:

means (110) for outputting said text in the detected text region only when a change is detected as to said text of said captured imagery.

43. The apparatus of claim 35, further comprising:

means (110) for outputting only said text within said agglomeration structure when said text is not detected in a current frame.

44. Method for portably recognizing text in a captured imagery, said method comprising the steps of:

(a) capturing an imagery having text information using a portable device;

(b) portably detecting a text region in the captured imagery in real time;

(c) applying optical character recognition (OCR) processing to said detected text region to produce recognized text; and

(d) providing said recognized text as an output of said portable device.

45. The method of claim 44, wherein said providing step (d) provides said output via a display.

46. The method of claim 44, wherein said providing step (d) provides said output via a speaker.

47. The method of claim 44, wherein said providing step (d) provides said output via an output port.

48. The method of claim 44, further comprising the step of:

(e) correlating supplemental information in accordance with said recognized text.

49. The method of claim 48, further comprising the step of:

(f) providing said supplemental information as an output of said portable device.

50. The method of claim 48, wherein said supplemental information contains travel information.

51. The method of claim 48, wherein said supplemental information contains vehicle information.

52. The method of claim 48, wherein said supplemental information contains information obtained from a web address.

53. The method of claim 48, further comprising the step of:
(f) dynamically applying lexicon processing in accordance with the correlated supplemental information.

54. The method of claim 44, further comprising the step of:
(e) applying language translation in accordance with said recognized text.

55. The method of claim 44, further comprising the step of:
(b1) adjusting said detected text region to produce a rectified image prior to the application of OCR processing.

56. The method of claim 55, further comprising the step of:
(b2) applying binarization to said rectified image prior to the application of OCR processing.

57. The method of claim 44, further comprising the step of:
(c1) applying agglomeration processing subsequent to said OCR processing to produce said recognized text.

58. The method of claim 44, further comprising the step of:
(c1) applying lexicon processing subsequent to said OCR processing to produce said recognized text.

59. The method of claim 58, wherein said lexicon processing is dynamically applied.

60. The method of claim 44, further comprising the step of:
(c1) applying false text elimination processing subsequent to said OCR processing to produce said recognized text.

61. The method of claim 44, further comprising the step of:
(e) providing said recognized text to a navigation system.

62. Apparatus (1200) for portably recognizing text in a captured imagery, said apparatus comprising:
an image capturing sensor (1216) for capturing an imagery having text information using a portable device;
a text detection module (1221) for portably detecting a text region in the captured imagery in real time;
an optical character recognition (OCR) module (1224) for applying OCR processing to said detected text region to produce recognized text; and
an output device (1210) for providing said recognized text as an output of said portable device.

63. The apparatus of claim 62, wherein said output device is a display (1215).

64. The apparatus of claim 62, wherein said output device is a speaker (1214).

65. The apparatus of claim 62, wherein said output device is an output port (1218).

66. The apparatus of claim 62, further comprising:
means (1230) for correlating supplemental information in accordance with said recognized text.

67. The apparatus of claim 66, wherein said output device further provides said supplemental information as an output of said portable device.

68. The apparatus of claim 66, wherein said supplemental information contains travel information.

69. The apparatus of claim 67, wherein said supplemental information contains vehicle information.

70. The apparatus of claim 66, further comprising:

a transmitter (1213) coupled to said correlating means; and

a receiver (1212) coupled to said output device, wherein said supplemental information contains information obtained from a web address.

71. The apparatus of claim 62, further comprising an application software module (1250) for applying language translation in accordance with said recognized text.

72. The apparatus of claim 62, further comprising a text orientation module (1222) for adjusting said detected text region to produce a rectified image prior to the application of OCR processing.

73. The apparatus of claim 72, further comprising a text binarization module (1223) for applying binarization to said rectified image prior to the application of OCR processing.

74. The apparatus of claim 62, further comprising an agglomeration module (1225) for applying agglomeration processing subsequent to said OCR processing to produce said recognized text.

75. The apparatus of claim 62, further comprising a lexicon module (1226) for applying lexicon processing subsequent to said OCR processing to produce said recognized text.

76. The apparatus of claim 62, further comprising a false detection module (1227) for applying false text elimination subsequent to said OCR processing to produce said recognized text.

77. The apparatus of claim 62, wherein said output device provides said recognized text to a navigation system.

78. Apparatus (1200) for portably recognizing text in a captured imagery, said apparatus comprising:

means (1216) for capturing an imagery having text information using a portable device;

means (1221) for portably detecting a text region in the captured imagery in real time;

means (1224) for applying optical character recognition (OCR) processing to said detected text region to produce recognized text; and

means (1210) for providing said recognized text as an output of said portable device.

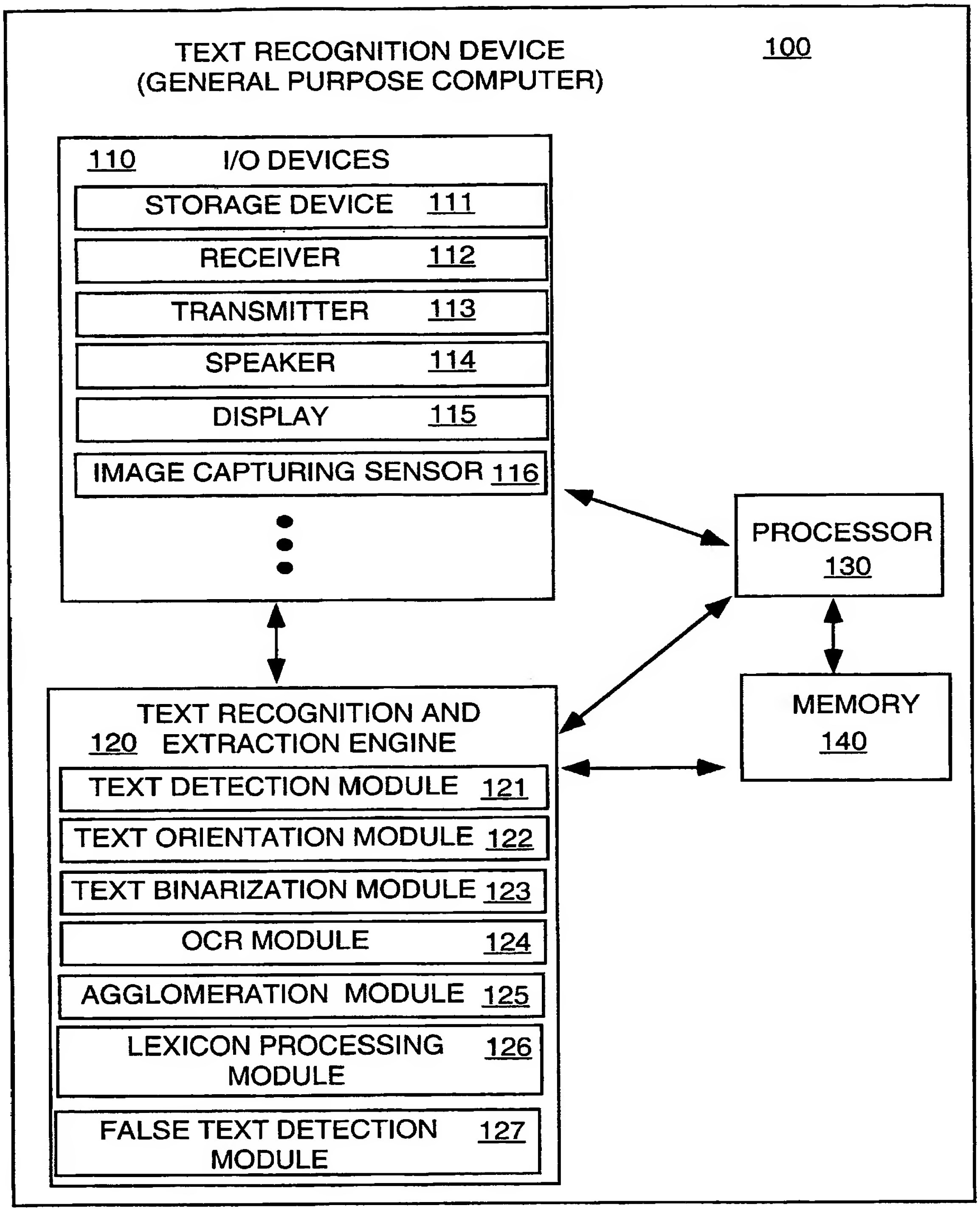

TEXT RECOGNITION DEVICE
(GENERAL PURPOSE COMPUTER)
100
110 I/O DEVICES
STORAGE DEVICE 111
RECEIVER 112
TRANSMITTER 113
SPEAKER 114
DISPLAY 115
IMAGE CAPTURING SENSOR 116
PROCESSOR 130
MEMORY 140
TEXT RECOGNITION AND
120 EXTRACTION ENGINE
TEXT DETECTION MODULE 121
TEXT ORIENTATION MODULE 122
TEXT BINARIZATION MODULE 123
OCR MODULE 124
AGGLOMERATION MODULE 125
LEXICON PROCESSING MODULE 126
FALSE TEXT DETECTION MODULE 127

FIG. 1

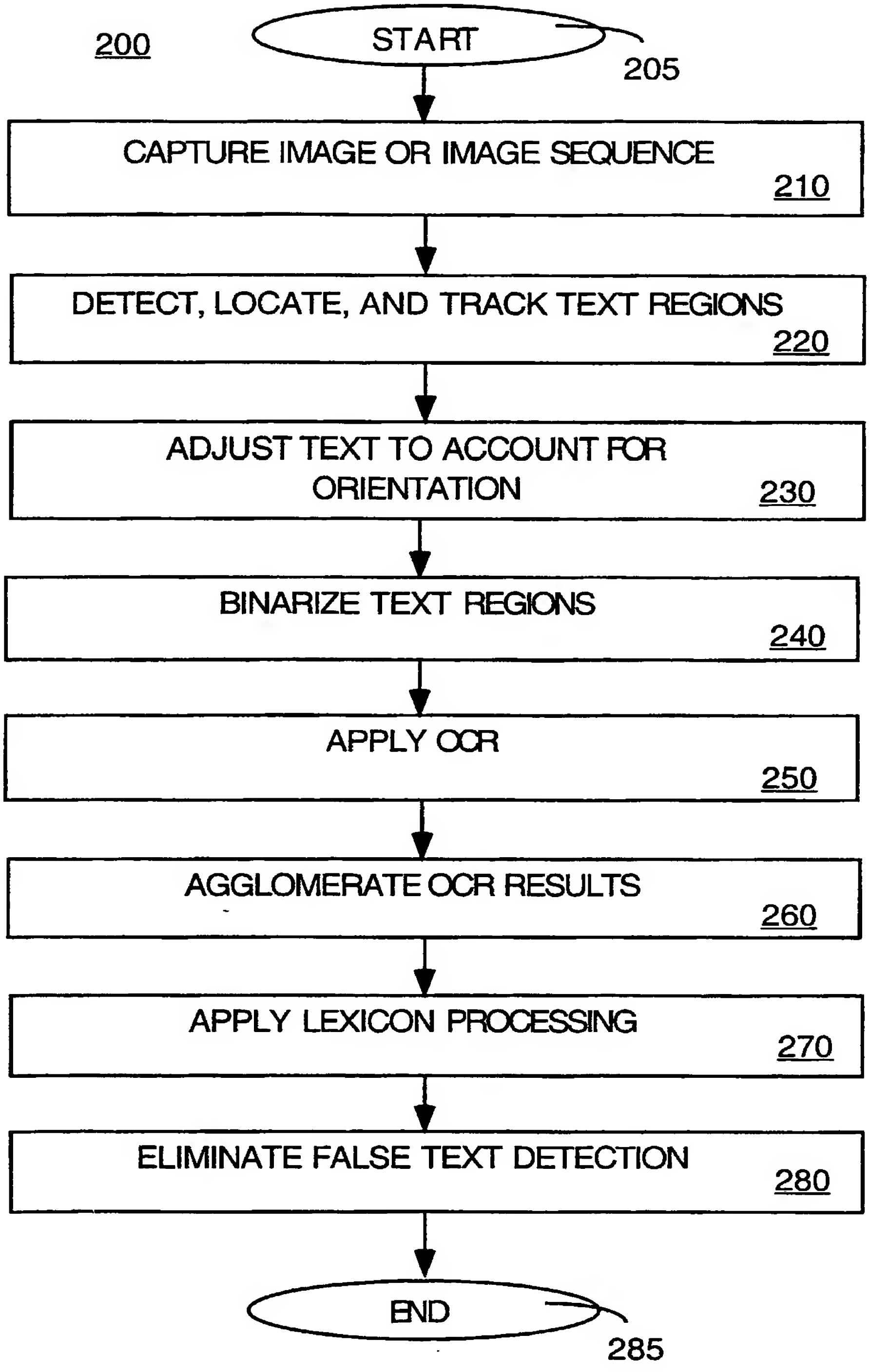

200
START
205
CAPTURE IMAGE OR IMAGE SEQUENCE
210
DETECT, LOCATE, AND TRACK TEXT REGIONS
220
ADJUST TEXT TO ACCOUNT FOR ORIENTATION
230
BINARIZE TEXT REGIONS
240
APPLY OCR
250
AGGLOMERATE OCR RESULTS
260
APPLY LEXICON PROCESSING
270
ELIMINATE FALSE TEXT DETECTION
280
END
285

FIG. 2

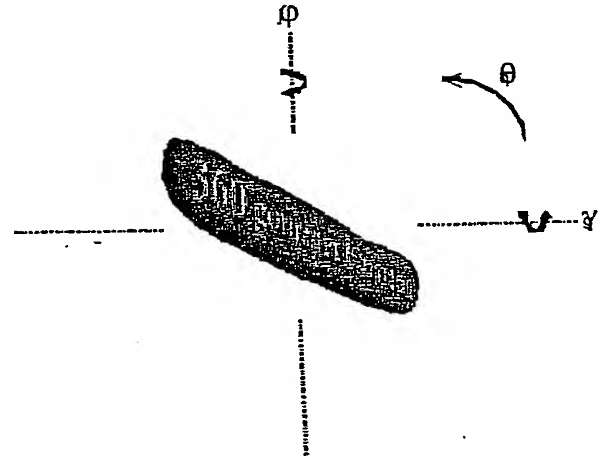


Figure 3

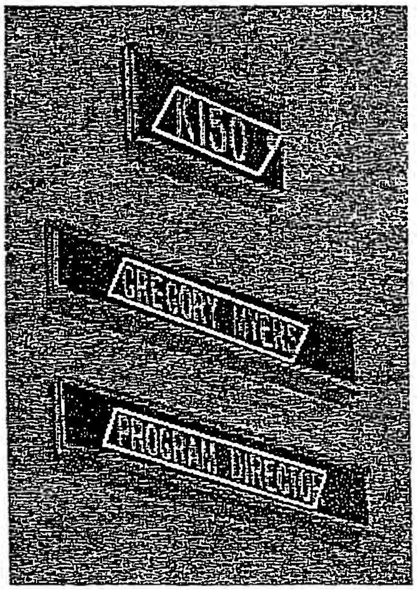


Figure 4

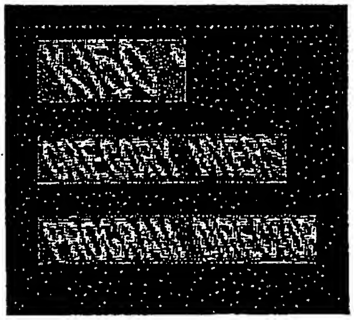


Figure 5

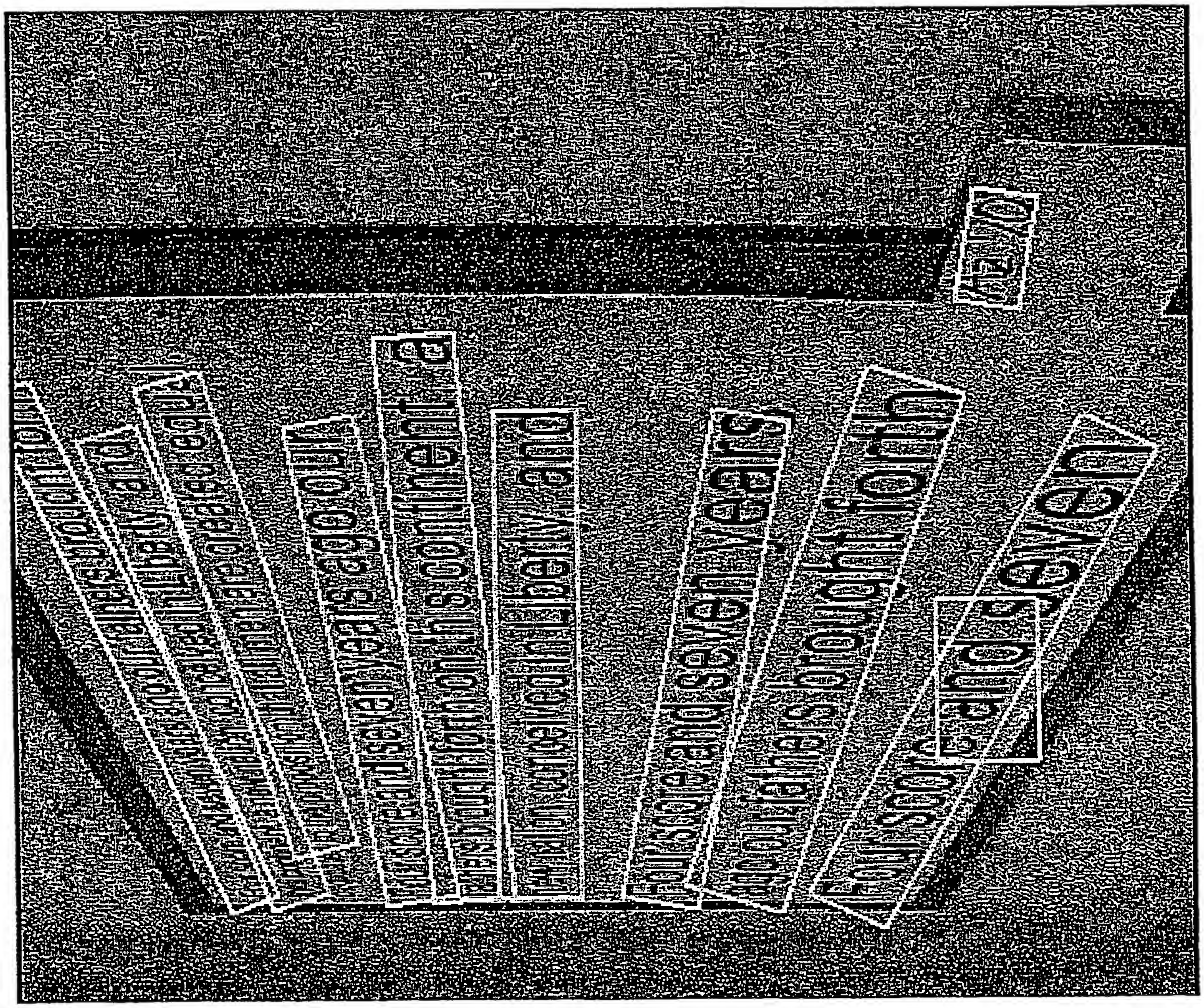

Four score and seven years
our fathers brought forth
Four score and seven
seven
Liberty, and
continent a

Fig 6

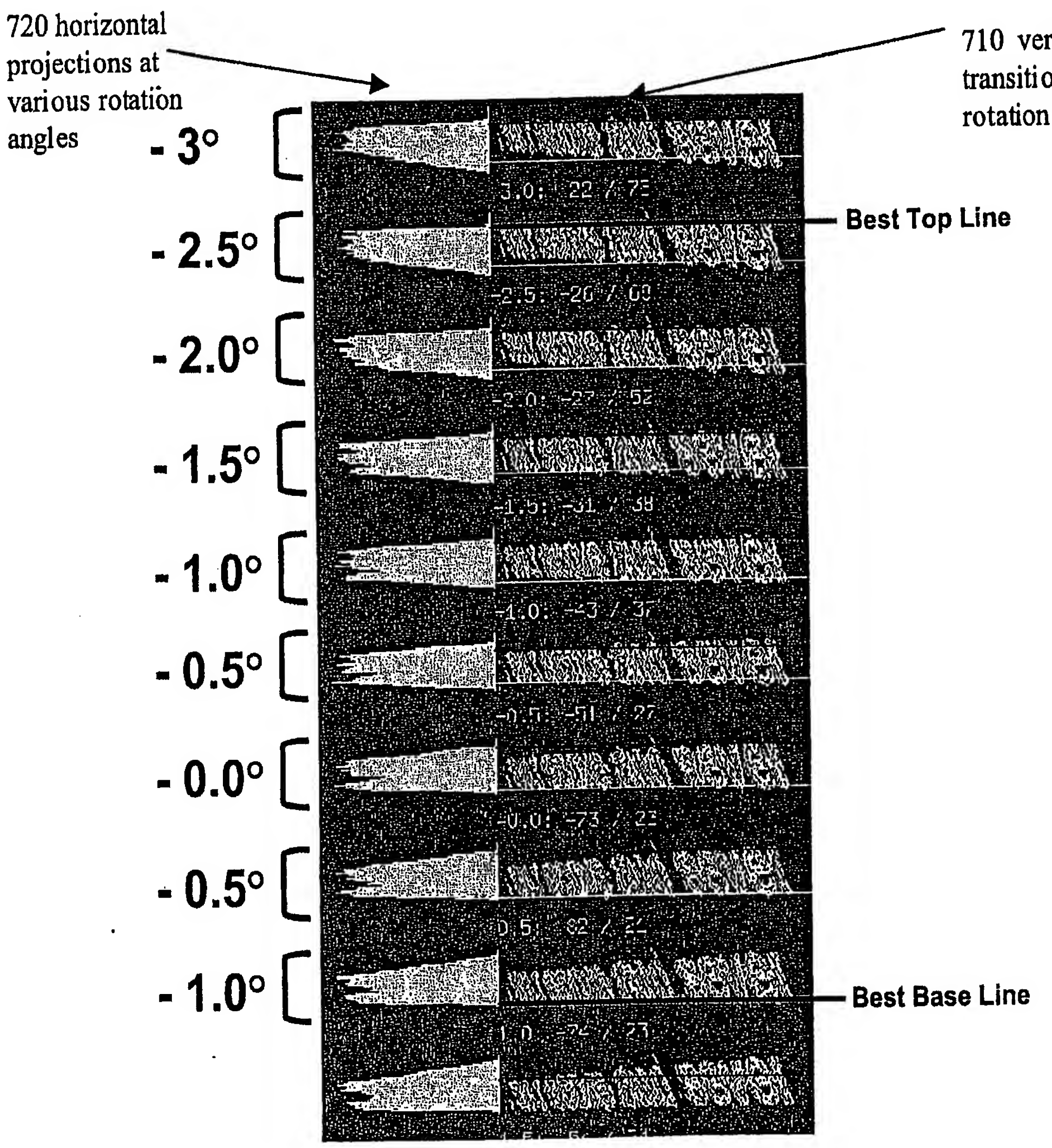

720 horizontal projections at various rotation angles
710 vertical edge transitions at various rotation angles
- 3°
- 2.5°
- 2.0°
- 1.5°
- 1.0°
- 0.5°
- 0.0°
- 0.5°
- 1.0°
3.0: 22 / 73
-2.5: -20 / 60
-2.0: -27 / 52
-1.5: -31 / 38
-1.0: -43 / 37
-0.5: -51 / 27
-0.0: -73 / 23
0.5: 82 / 22
1.0: -74 / 23
Best Top Line
Best Base Line

Fig 7

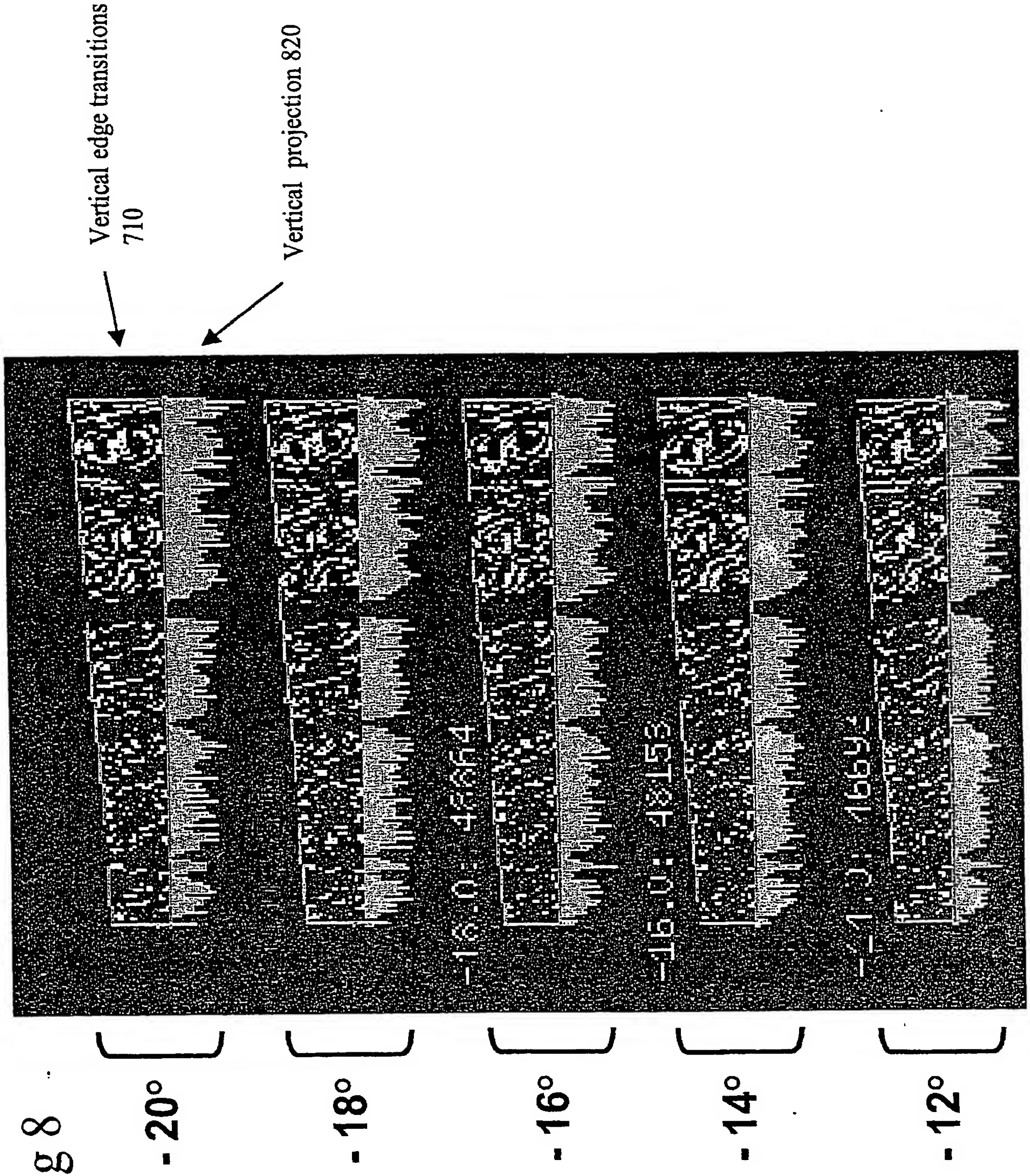

Vertical edge transitions 710
Vertical projection 820
-20°
-18°
-16°
-14°
-12°

Fig 8

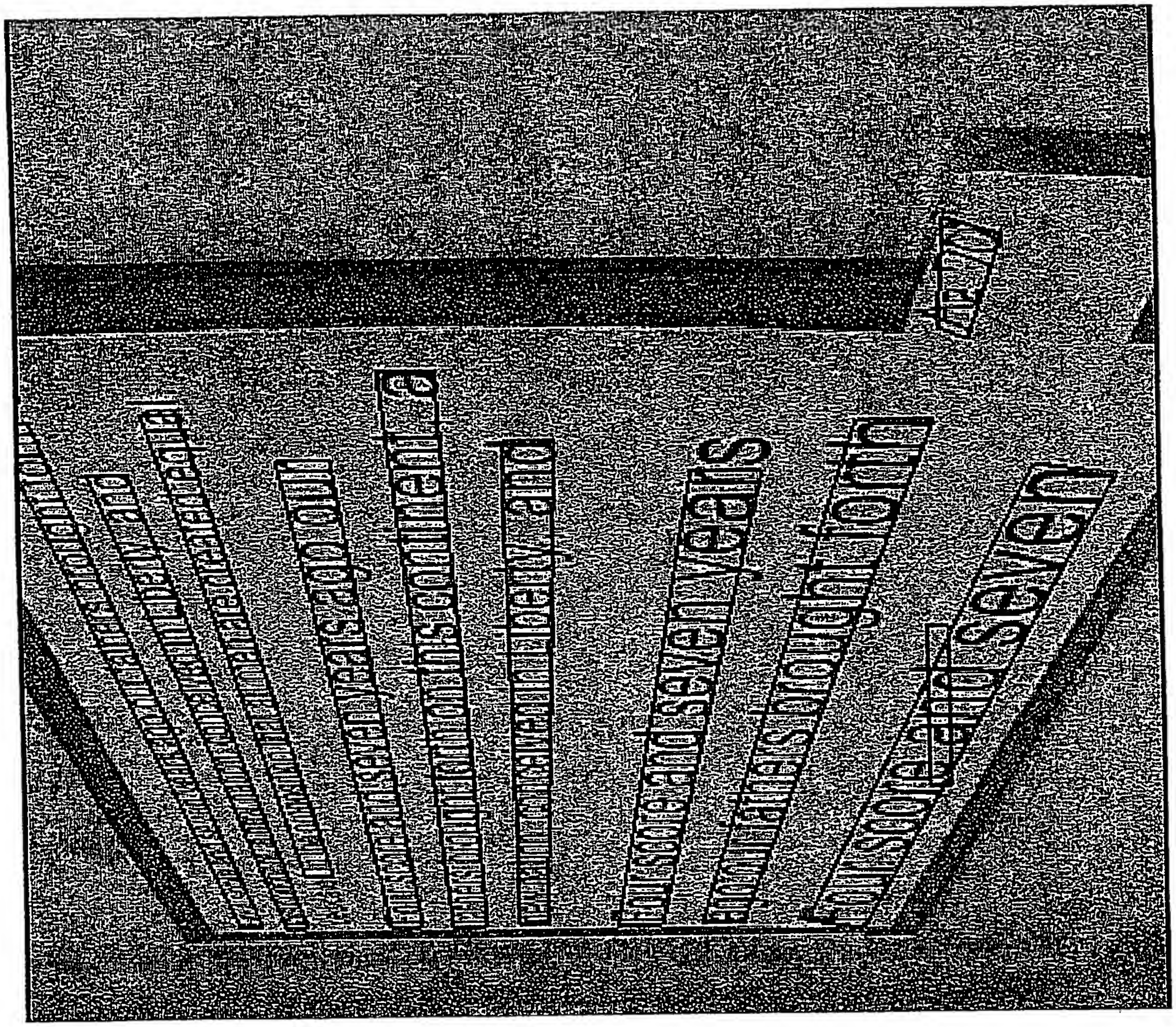
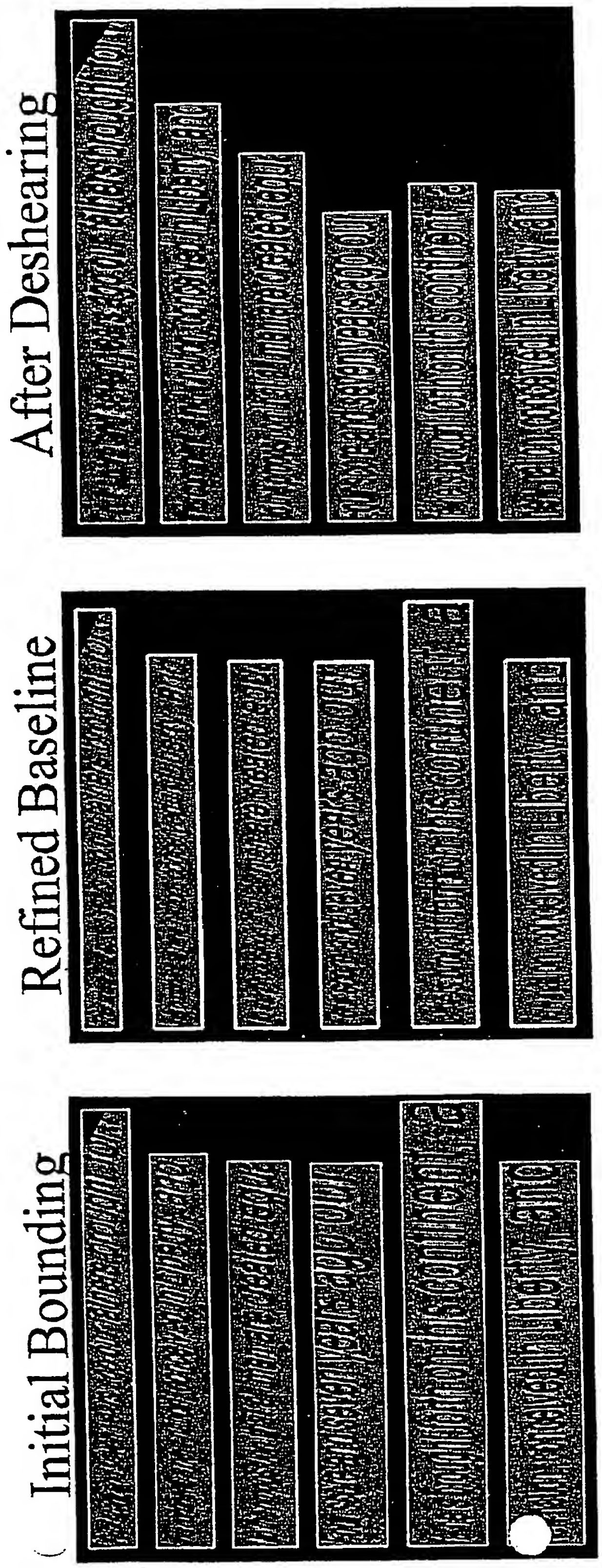

Four score and seven
Four score and seven years
our fathers brought forth
Apply

Fig 9

Fig 10



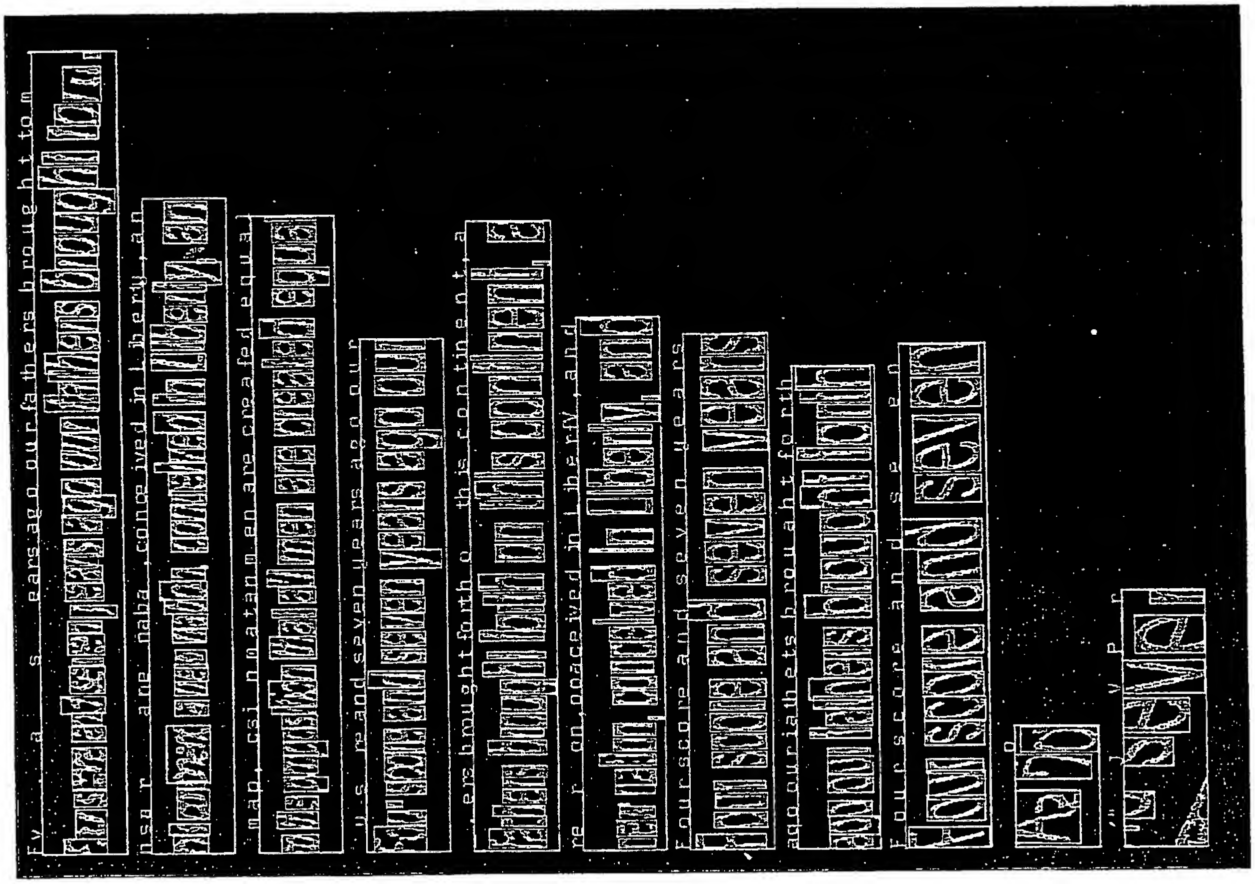

F o u r s c o r e a n d s e v e n y e a r s
ago our fathers brought forth
F o u r s c o r e a n d s e v e n

Fig 11

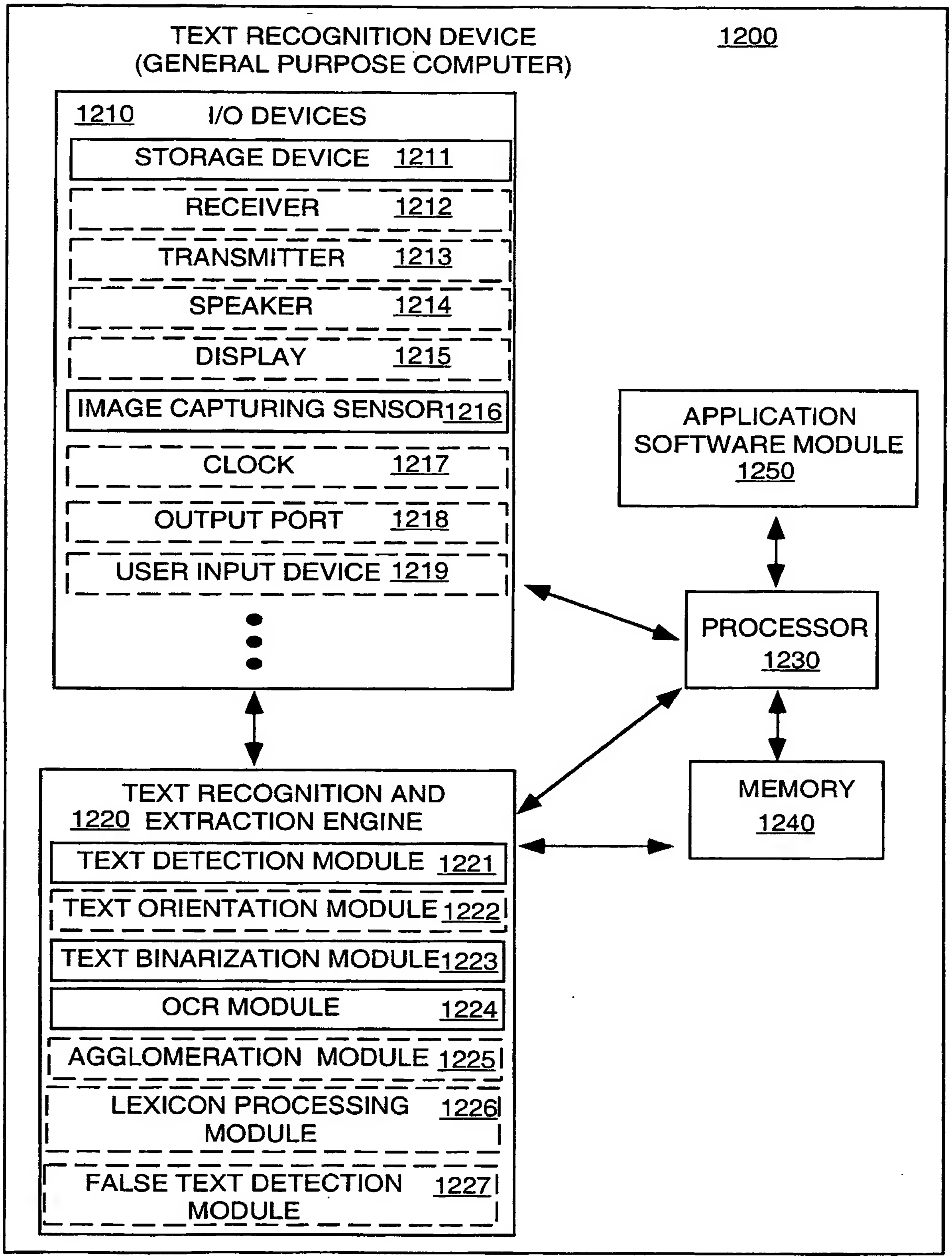

TEXT RECOGNITION DEVICE
(GENERAL PURPOSE COMPUTER)
1200
1210 I/O DEVICES
STORAGE DEVICE 1211
RECEIVER 1212
TRANSMITTER 1213
SPEAKER 1214
DISPLAY 1215
IMAGE CAPTURING SENSOR 1216
CLOCK 1217
OUTPUT PORT 1218
USER INPUT DEVICE 1219
APPLICATION SOFTWARE MODULE 1250
PROCESSOR 1230
MEMORY 1240
TEXT RECOGNITION AND
1220 EXTRACTION ENGINE
TEXT DETECTION MODULE 1221
TEXT ORIENTATION MODULE 1222
TEXT BINARIZATION MODULE 1223
OCR MODULE 1224
AGGLOMERATION MODULE 1225
LEXICON PROCESSING MODULE 1226
FALSE TEXT DETECTION MODULE 1227

FIG. 12

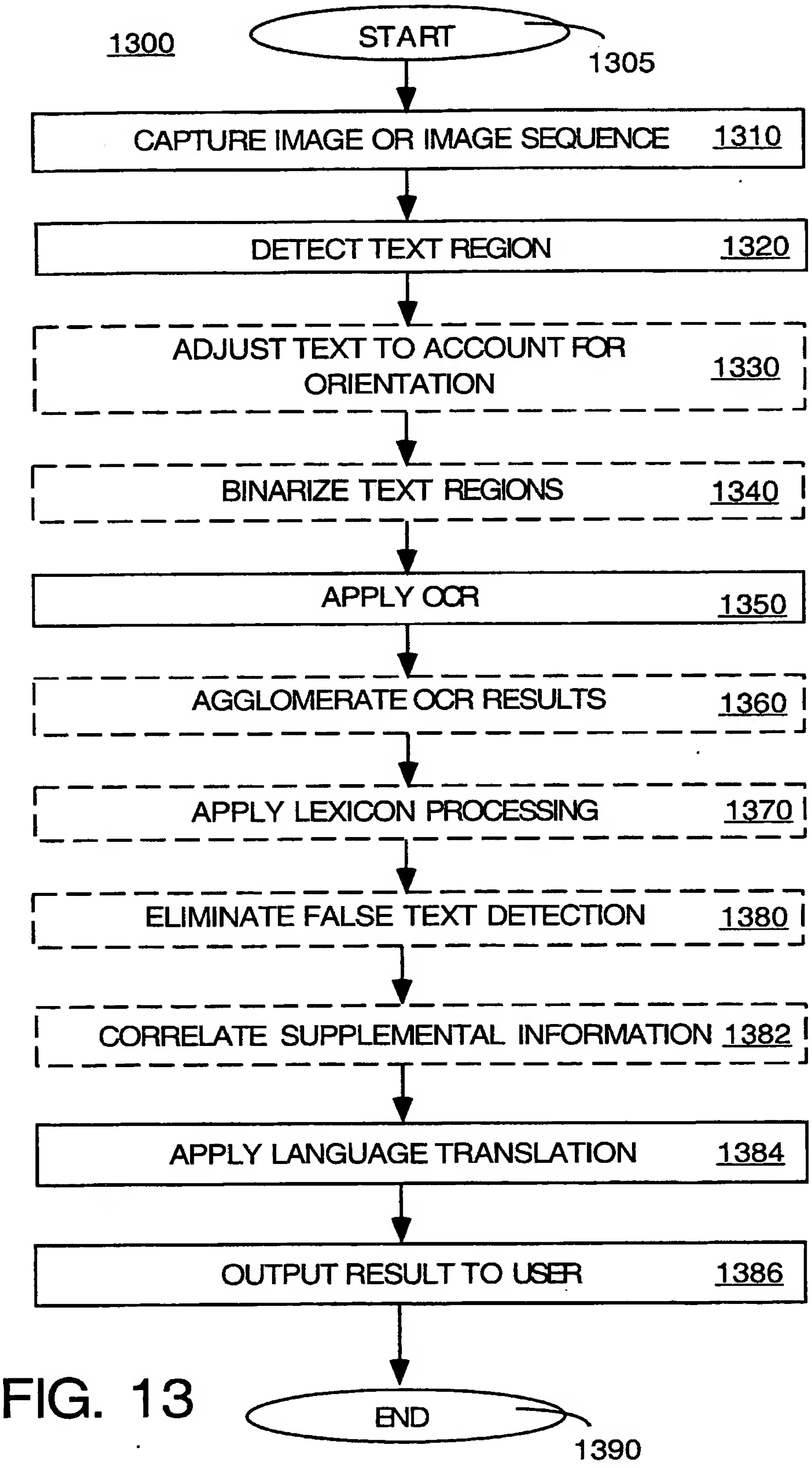

1300
START
1305
CAPTURE IMAGE OR IMAGE SEQUENCE 1310
DETECT TEXT REGION 1320
ADJUST TEXT TO ACCOUNT FOR ORIENTATION 1330
BINARIZE TEXT REGIONS 1340
APPLY OCR 1350
AGGLOMERATE OCR RESULTS 1360
APPLY LEXICON PROCESSING 1370
ELIMINATE FALSE TEXT DETECTION 1380
CORRELATE SUPPLEMENTAL INFORMATION 1382
APPLY LANGUAGE TRANSLATION 1384
OUTPUT RESULT TO USER 1386
END
1390

FIG. 13

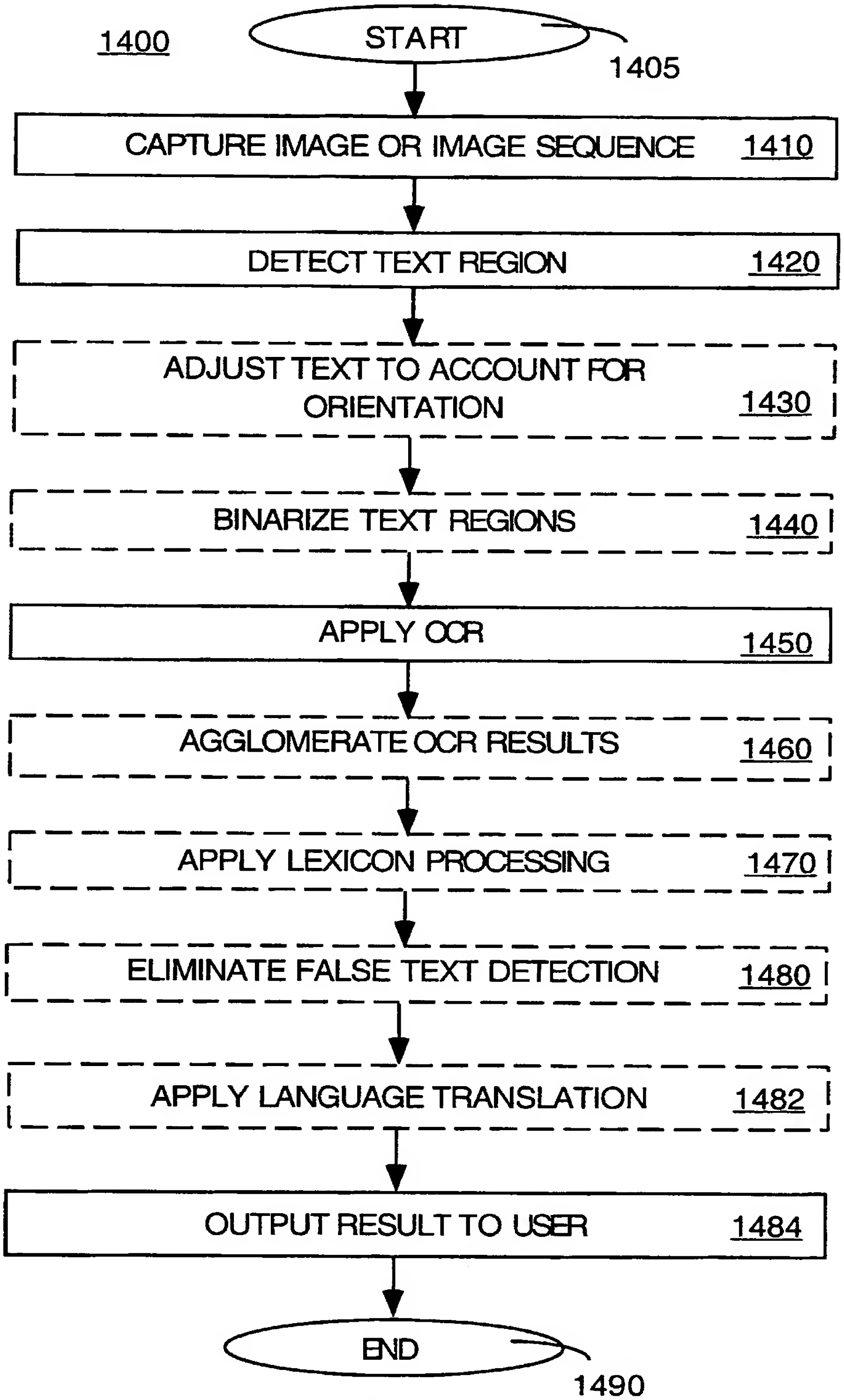

1400
START
1405
CAPTURE IMAGE OR IMAGE SEQUENCE 1410
DETECT TEXT REGION 1420
ADJUST TEXT TO ACCOUNT FOR ORIENTATION 1430
BINARIZE TEXT REGIONS 1440
APPLY OCR 1450
AGGLOMERATE OCR RESULTS 1460
APPLY LEXICON PROCESSING 1470
ELIMINATE FALSE TEXT DETECTION 1480
APPLY LANGUAGE TRANSLATION 1482
OUTPUT RESULT TO USER 1484
END
1490

FIG. 14

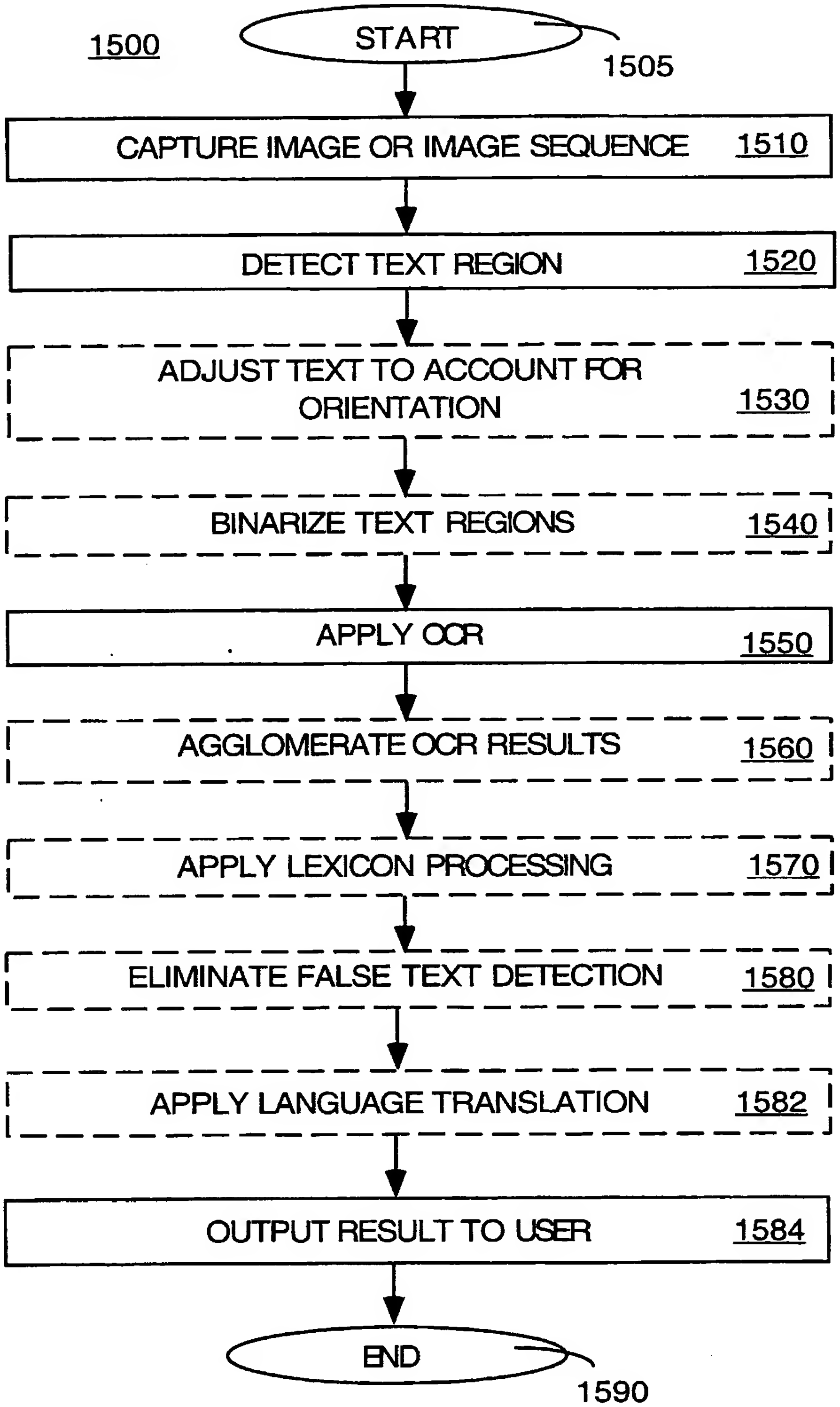

1500
START
1505
CAPTURE IMAGE OR IMAGE SEQUENCE 1510
DETECT TEXT REGION 1520
ADJUST TEXT TO ACCOUNT FOR ORIENTATION 1530
BINARIZE TEXT REGIONS 1540
APPLY OCR 1550
AGGLOMERATE OCR RESULTS 1560
APPLY LEXICON PROCESSING 1570
ELIMINATE FALSE TEXT DETECTION 1580
APPLY LANGUAGE TRANSLATION 1582
OUTPUT RESULT TO USER 1584
END
1590

FIG. 15

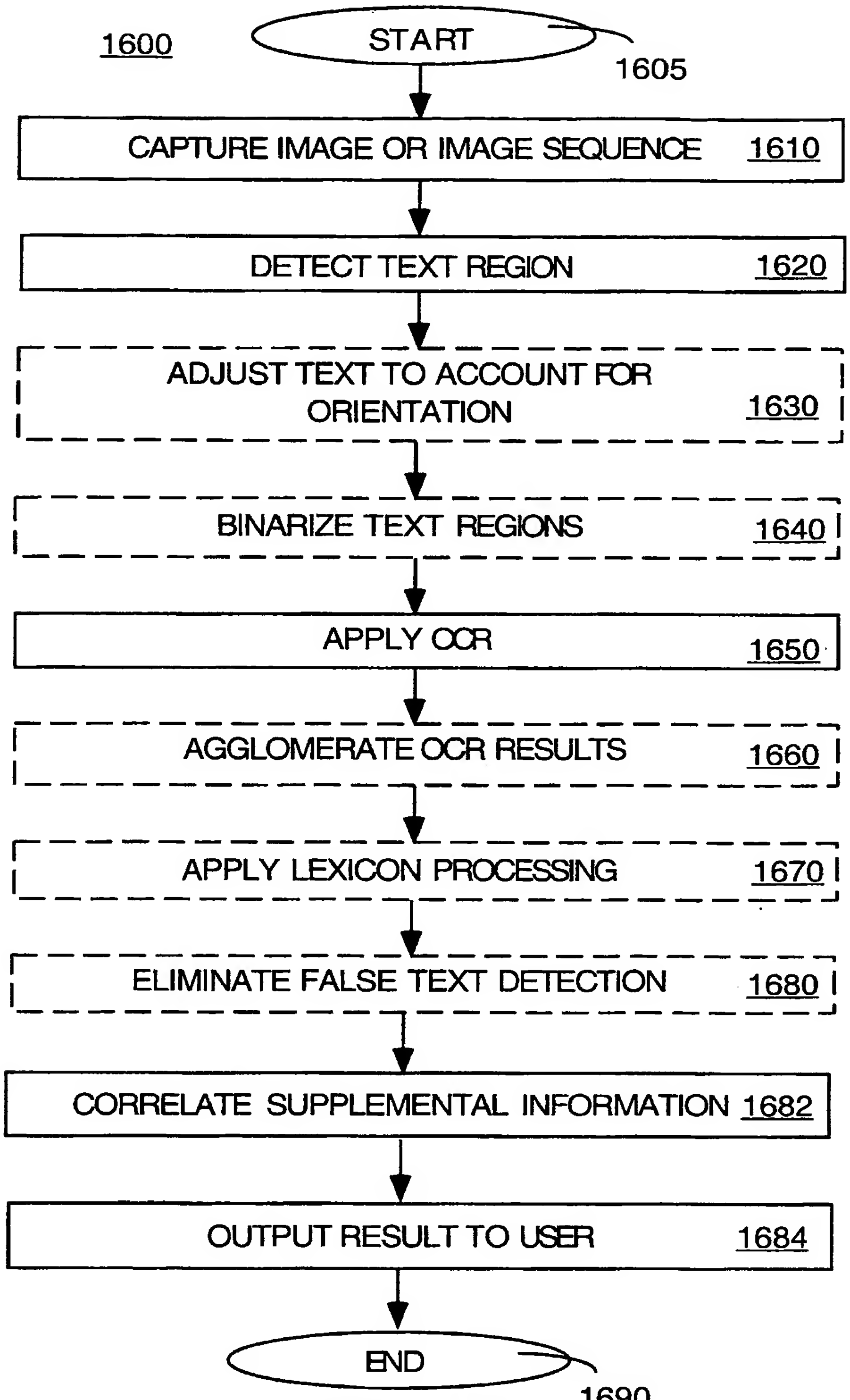

1600
START
1605
CAPTURE IMAGE OR IMAGE SEQUENCE 1610
DETECT TEXT REGION 1620
ADJUST TEXT TO ACCOUNT FOR ORIENTATION 1630
BINARIZE TEXT REGIONS 1640
APPLY OCR 1650
AGGLOMERATE OCR RESULTS 1660
APPLY LEXICON PROCESSING 1670
ELIMINATE FALSE TEXT DETECTION 1680
CORRELATE SUPPLEMENTAL INFORMATION 1682
OUTPUT RESULT TO USER 1684
END
1690

FIG. 16

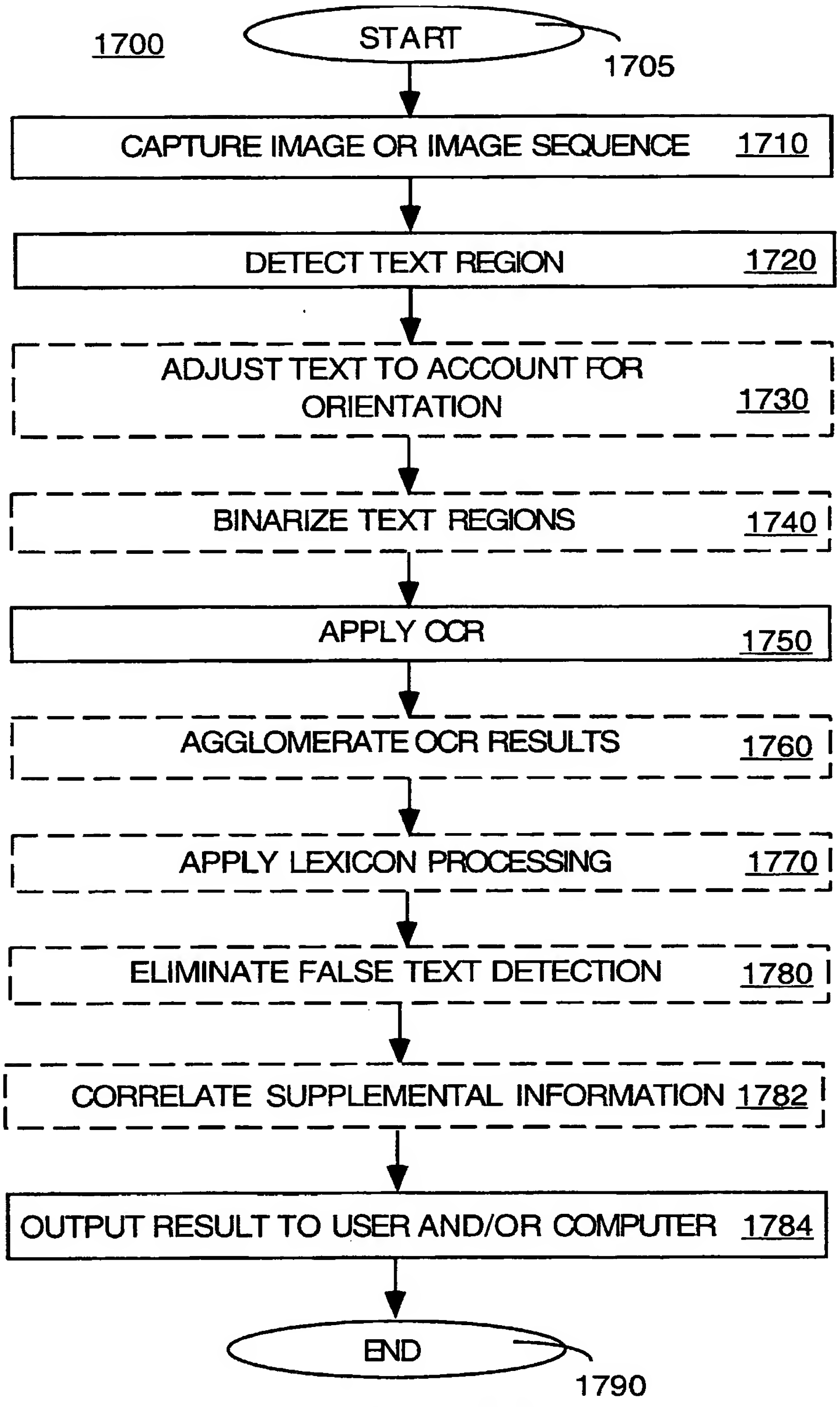

1700
START
1705
CAPTURE IMAGE OR IMAGE SEQUENCE 1710
DETECT TEXT REGION 1720
ADJUST TEXT TO ACCOUNT FOR ORIENTATION 1730
BINARIZE TEXT REGIONS 1740
APPLY OCR 1750
AGGLOMERATE OCR RESULTS 1760
APPLY LEXICON PROCESSING 1770
ELIMINATE FALSE TEXT DETECTION 1780
CORRELATE SUPPLEMENTAL INFORMATION 1782
OUTPUT RESULT TO USER AND/OR COMPUTER 1784
END
1790

FIG. 17

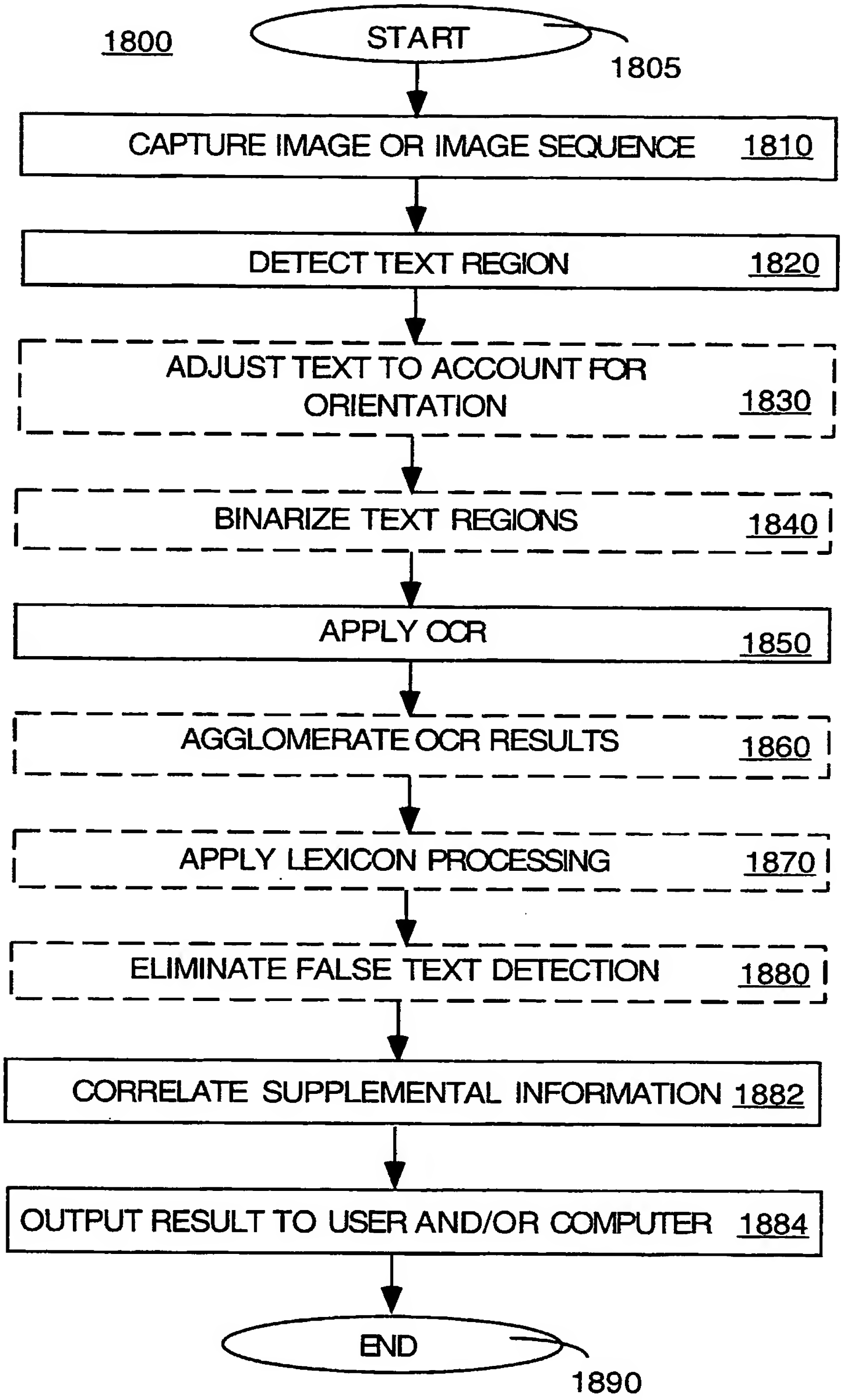

1800
START
1805
CAPTURE IMAGE OR IMAGE SEQUENCE 1810
DETECT TEXT REGION 1820
ADJUST TEXT TO ACCOUNT FOR ORIENTATION 1830
BINARIZE TEXT REGIONS 1840
APPLY OCR 1850
AGGLOMERATE OCR RESULTS 1860
APPLY LEXICON PROCESSING 1870
ELIMINATE FALSE TEXT DETECTION 1880
CORRELATE SUPPLEMENTAL INFORMATION 1882
OUTPUT RESULT TO USER AND/OR COMPUTER 1884
END
1890

FIG. 18

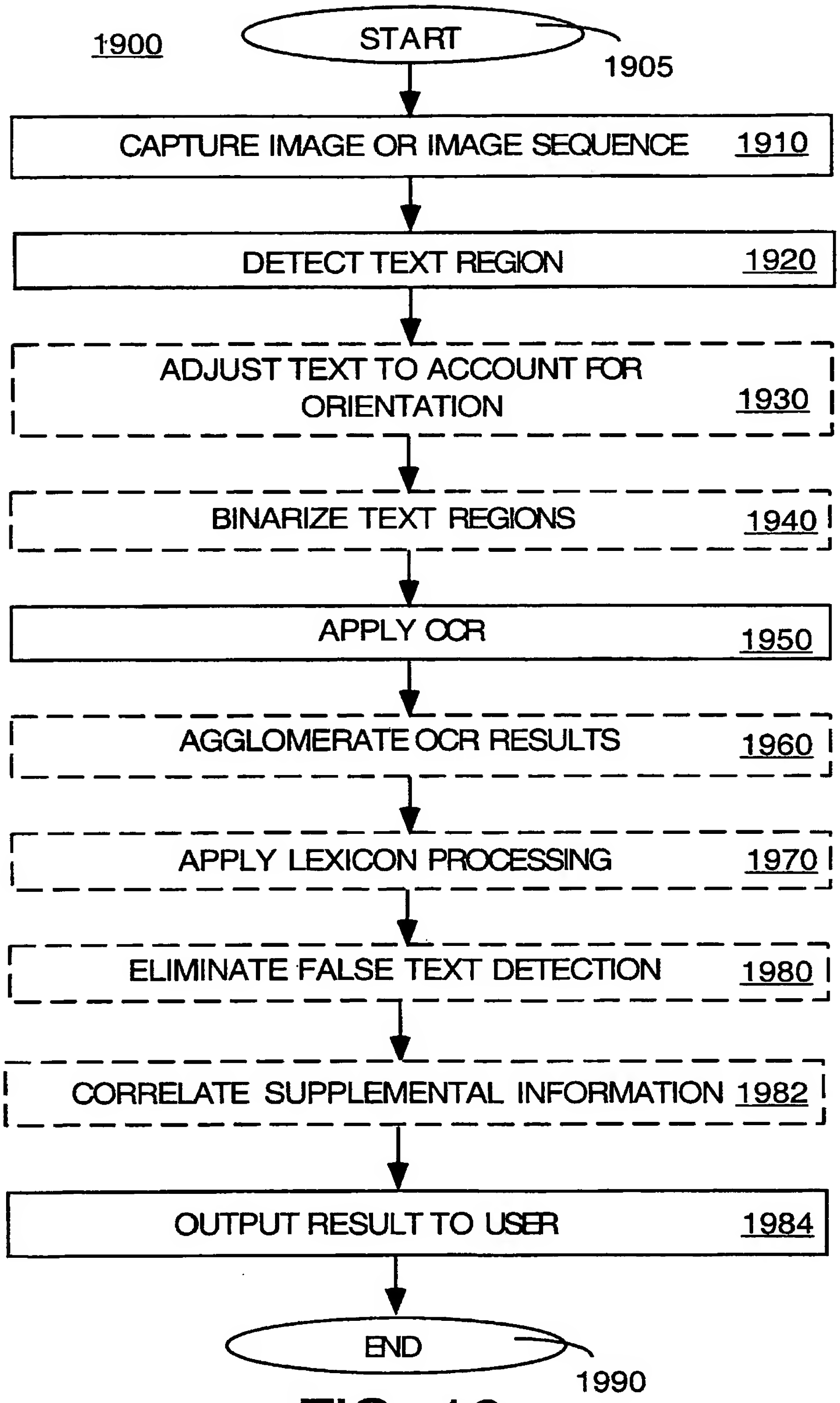

1900
START
1905
CAPTURE IMAGE OR IMAGE SEQUENCE 1910
DETECT TEXT REGION 1920
ADJUST TEXT TO ACCOUNT FOR ORIENTATION 1930
BINARIZE TEXT REGIONS 1940
APPLY OCR 1950
AGGLOMERATE OCR RESULTS 1960
APPLY LEXICON PROCESSING 1970
ELIMINATE FALSE TEXT DETECTION 1980
CORRELATE SUPPLEMENTAL INFORMATION 1982
OUTPUT RESULT TO USER 1984
END
1990

FIG. 19